

# System Viewers CSCI

## Design Review

### Thor DP3

84K00540-010

November 19, 1997  
Version 1.0

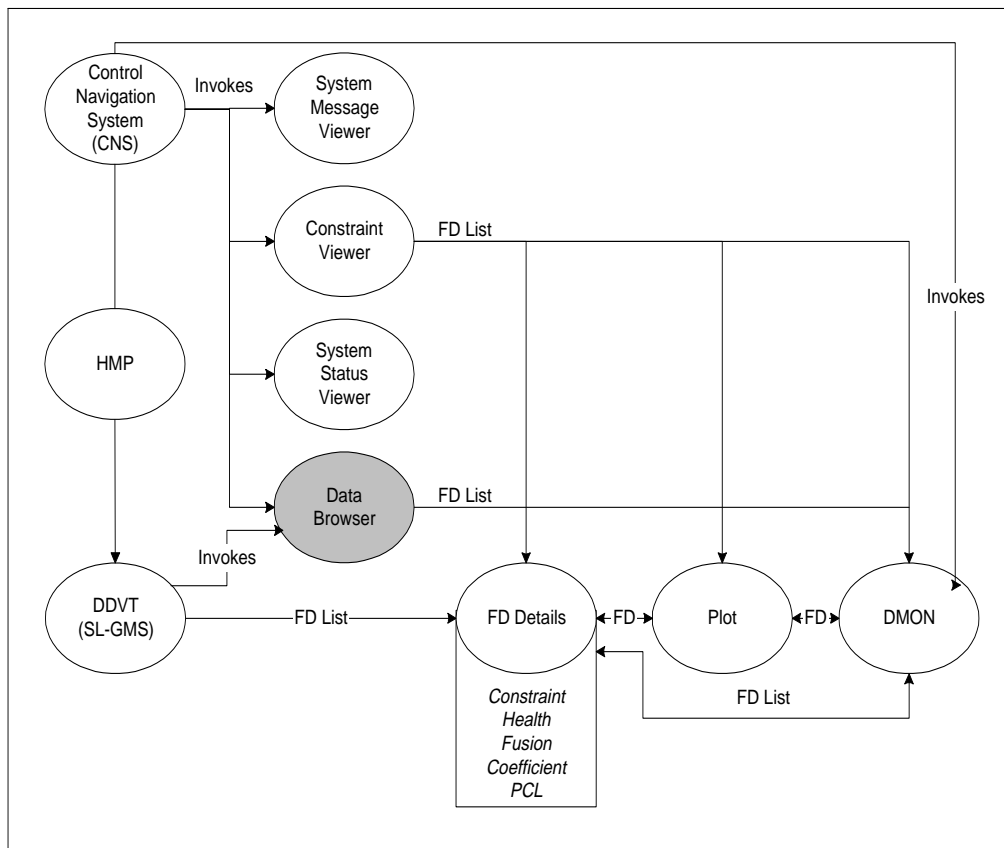
## 1. Data Browser

### 1.1 Data Browser Introduction

#### 1.1.1 Data Browser Overview

The Data Browser provides the Checkout and Launch Control System (CLCS) with a method for selecting a Function Designator (FD) from a list of FDs. A list of FDs may be selected.

The following diagram shows the relationship of the Data Browser to the other components of the System Viewers.



#### 1.1.2 Data Browser Operational Description

The Data Browser is invoked from the Control Navigation System (CNS) or the DDVT. A “pop-up” window that contains a list of the Function Designator (FD) choices for the user will be displayed. A Data Browser choice is made by highlighting a Data Browser item using the mouse, then releasing the mouse button while the item is highlighted. The user will select either FD Details, Plot, or DMON Viewer.

# Software Requirements and Design Specification

## 1.2 Data Browser Specifications

### 1.2.1 Data Browser Ground Rules

1. For the Thor delivery the list of FDs that are displayed for selection will be a subset of all possible FDs (e.g. User Class).
2. In order to run on the BASIS environment the On-Line Data Bank and System Configuration Table must be available. The BASIS environment will provide monitoring only capability.

### 1.2.2 Data Browser Functional Requirements

1. The Data Browser will provide a display to select an FD or group of FDs. Once the FD is selected the user may elect to view the FD in either the FD Details, DMON, or the Plot viewers.
2. The Data Browser will have a scrolled list of FDs from which the user may choose one FD by clicking to highlight.
3. The Data Browser will provide a capability to invoke the FD Details, DMON, and Plot viewers.
4. The Data Browser will provide a capability to allow the user to type in the desired FD using the keyboard.
5. The Data Browser will provide a capability to allow the user to type one or more characters and the Data Browser will position itself to the location in the FD list that begins with the character(s) that was typed in.
6. The Data Browser will conform to the "CLCS HCI Style Guide & Standards" document number 84K00230.
7. *The Data Browser will also execute in the BASIS environment.*

### 1.2.3 Data Browser Performance Requirements

- There is no performance requirements for the Data Browser.

### 1.2.4 Data Browser Interfaces

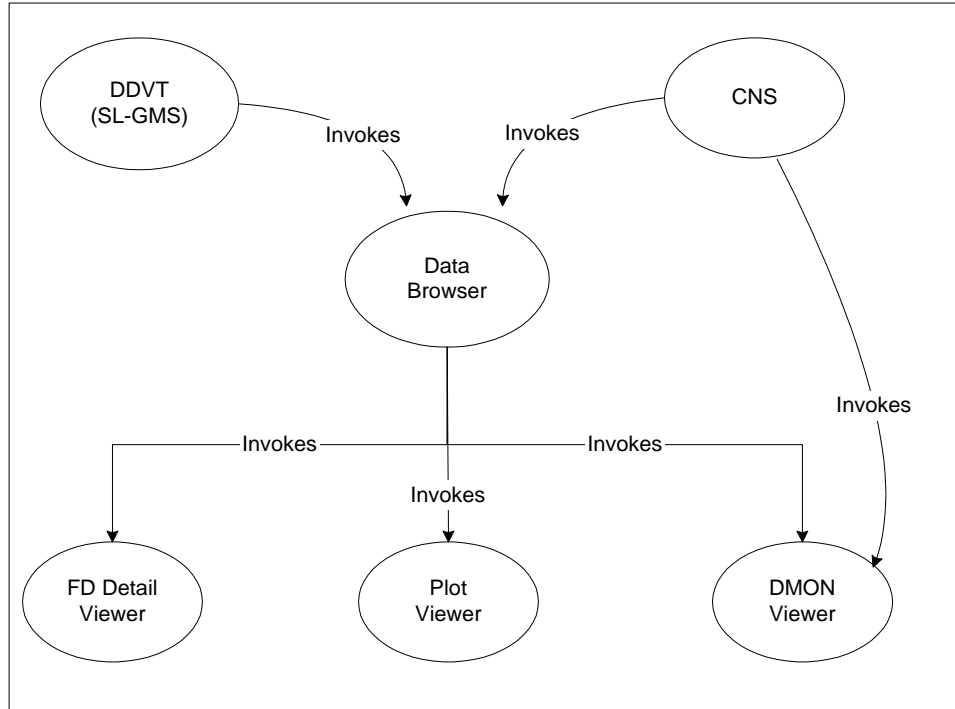
The Data Browser is invoked from the control navigation system or DDVT.

Once a Data Browser is created and a choice from the Data Browser is made, the Data Browser initiates a call to the appropriate System Viewer corresponding to the choice.

# Software Requirements and Design Specification

## 1.2.5 Data Browser Data Flow Diagram

The Data Browser is invoked from either the DDVT or CNS. The Data Browser will invoke either the FD Details, Plot Viewer, or the DMON Viewer.



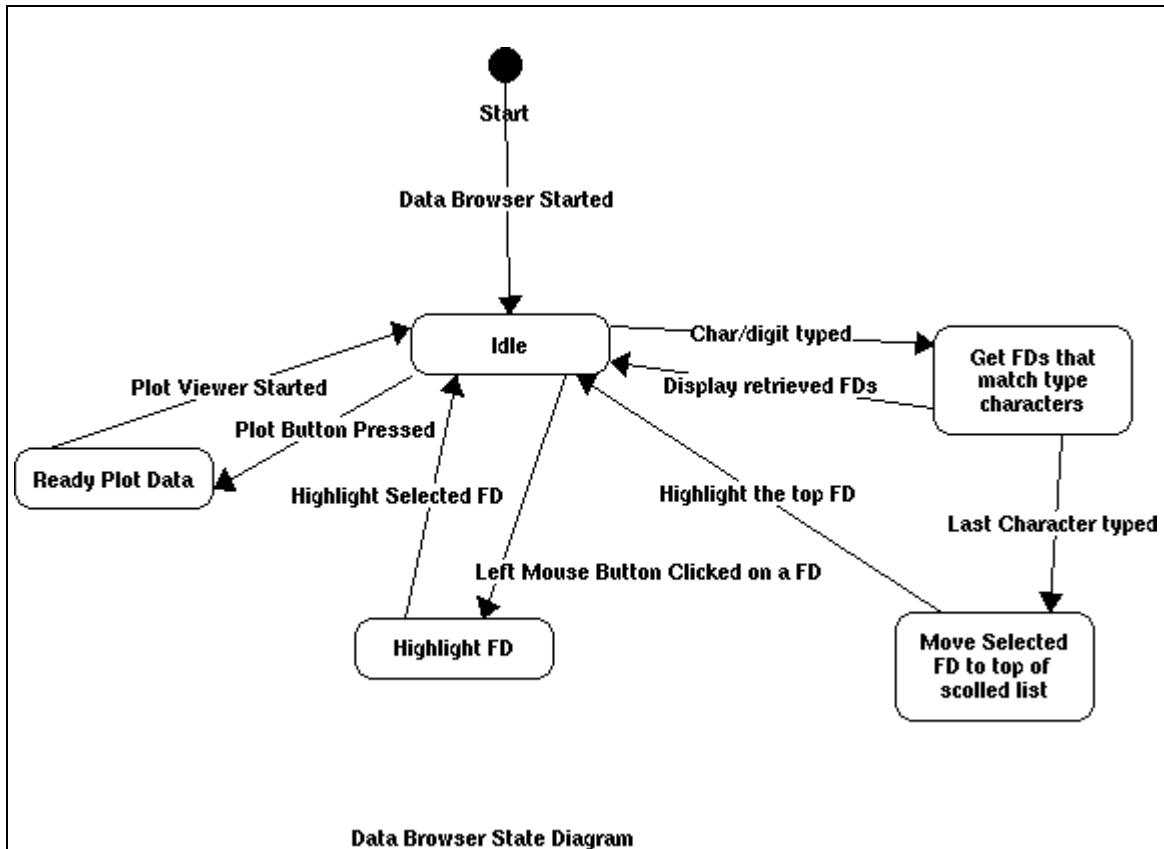
## 1.3 Data Browser Design Specification

The Data Browser runs in the CCWS, and is invoked from the DDVT (SL-GMS Application) or from the CNS. It is implemented in Java. Each element of variable information on the Data Browser screen is represented as a text field.

Appendix A contains a Class Diagram for the Data Browser.

## 1.3.1 Data Browser State Diagram

The Data Browser State Diagram shown below relates the states and the events of the Data Browser.



## 1.3.2 Data Browser External Interface

The Data Browser will not provide any APIs. It will call an API from the Data Handler within User Display Services. The Data Handler in UDS will send the FD Name and FD Nomenclature to the Data Browser. These interfaces are described in the User Display Services Interface Description Document (IDD) 84K00361.

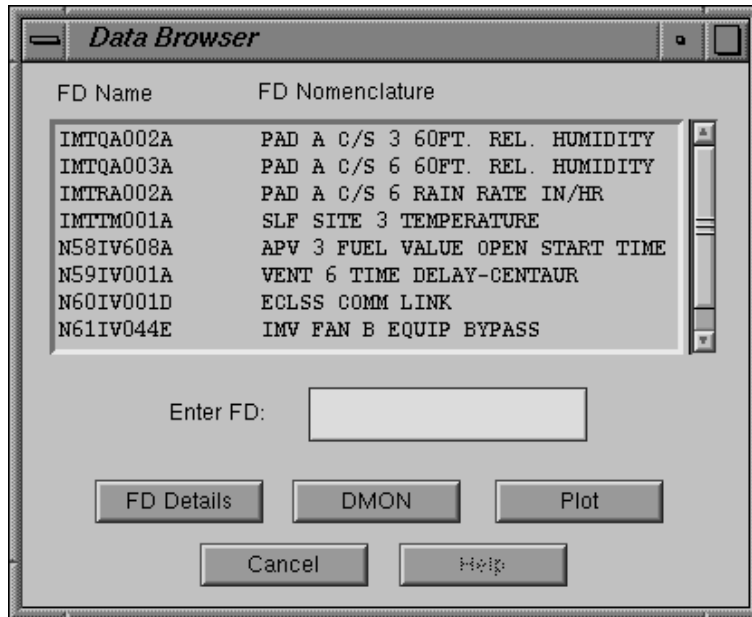
### 1.3.2.1 Data Browser Message Formats

Not applicable.

# Software Requirements and Design Specification

## 1.3.2.2 Data Browser Display Formats

The Data Browser is invoked via the DDVT (SL-GMS Application) or CNS.



**Figure 4. Main Window of Data Browser.**

### Main Window Functions

1. Display the FD Name and FD Nomenclature at the appropriate location listed in a scrolled window.
2. Allow users to enter additional FDs.
3. Allow users to search and scroll through a list of FDs.
4. Allow users to invoke other viewers including FD Details, DMON and Plot.
5. Allow users to close the Data Browser with a cancel button.
6. Allow users to invoke the Help Window with a help button.

## 1.3.2.3 Data Browser Input Formats

Not applicable.

## 1.3.2.4 Data Browser Recorded Data

Not applicable.

## 1.3.2.5 Data Browser Printer Formats

Not applicable.

# Software Requirements and Design Specification

## 1.3.2.6 Data Browser Inter-process Communications

The Data Browser receives data only from Application Services. The Data Browser will display the FD Name and FD Nomenclature.

## 1.3.2.7 Data Browser External Interface Calls

The Data Browser will not provide any APIs. It will call the following API from the Data Handler within User Display Services. The Data Handler in UDS will send the FD Name and FD Nomenclature to the Data Browser. These interfaces are described in the User Display Services Interface Description Document (IDD) 84K00361.

## 1.3.3 Data Browser Test Plan

The Data Browser system level tests may be run in either the IDE or SDE, or both. These tests are run on the CCWS platform. Application services will be required to send data to the Data Browser for FD Name and FD Nomenclature.

The specific test cases that will be run include:

1. Verify a FD Name and FD Nomenclature is received in the Data Browser Window.
2. Verify that one FD can be selected.
3. Verify that a group of FDs can be selected.
4. Verify print screen capability.
5. Verify the DMON Button invokes the DMON Viewer.
6. Verify the Plot Button invokes the Plot View.
7. Verify the FD Details Button invokes the FD Details Window.
8. Verify the Cancel Button closes the Data Browser.
9. *Verify the Help Button invokes the Help Window.*

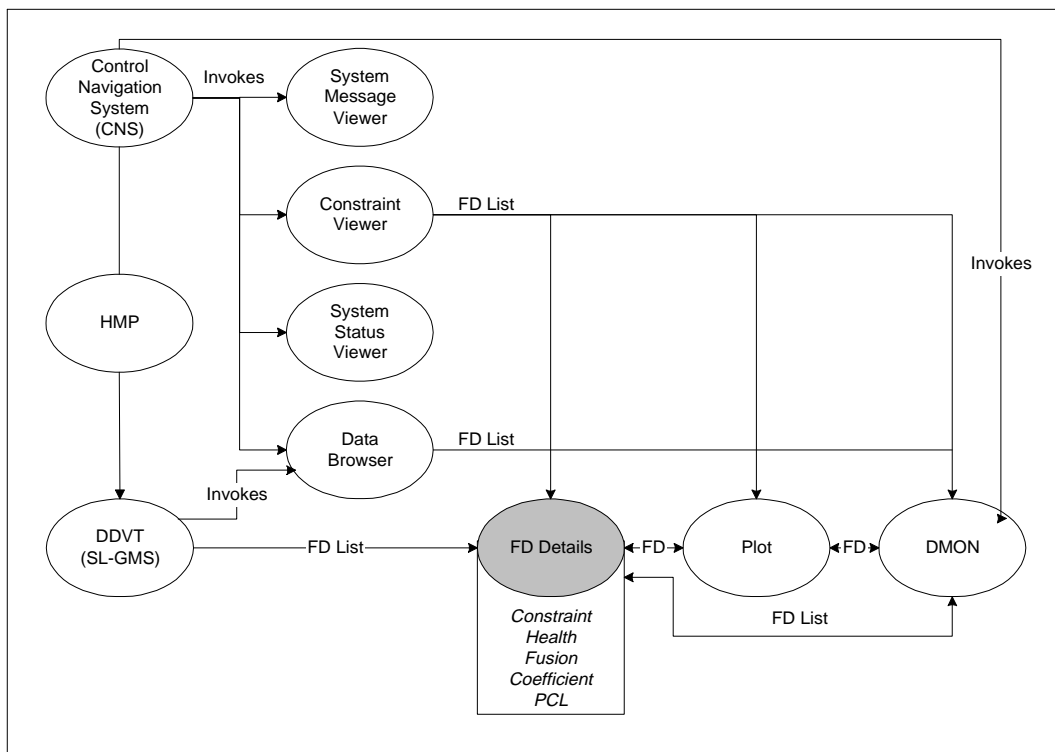
## 2. FD Details Viewer

### 2.1 FD Details Viewer Introduction

#### 2.1.1 FD Details Viewer Overview

The FD Details Viewer provides the Checkout and Launch Control System (CLCS) with the means to display on the Command and Control Workstation (CCWS) information pertaining to a Function Designator (FD). Data may be viewed in tabular format. Plotting support for data viewing is provided in the Plotting Viewer.

The diagram below shows the relationship of the FD Details Viewer to other components within the System Viewers.





# Software Requirements and Design Specification

## 2.1.2 FD Details Viewer Operational Description

The FD Details Viewer creates a window in which information about FDs are displayed. The information is presented as a rectangular box, with labels for each field. The FD Details Viewer may invoke the DMON Viewer or the Plot Viewer.

The FD Details Viewer provides the means to see additional details about an FD. Constraint, PCL, Data Fusion, Data Health, Change coefficients viewers are accessible in the FD Details Viewer. The FD Details window will expand to show the selected view.

## 2.2 FD Details Viewer Specifications

### 2.2.1 FD Details Viewer Ground Rules

1. Data for FD Details will be received from an API(s) from User Display Services.
2. Redstone Status FD is referred to as FD Details in the Thor delivery.
3. In order to run on the BASIS environment the On-Line Data Bank and System Configuration Table must be available. The BASIS environment will provide monitoring only capability.

### 2.2.2 FD Details Viewer Functional Requirements

1. The FD Details Viewer will provide a window to display a Function Designator (FD) on the CCWS.
2. The FD Details Viewer will display specific information about an FD in its window:
  - FD name
  - FD nomenclature
  - FDID
  - Time
  - Health
  - FD value (in engineering units where applicable)
  - Coefficient
  - Other details as specified by users/developers
3. ~~T~~The FD Details Viewer will provide a capability to invoke the DMON Viewer.
4. ~~T~~The FD Details Viewer will provide a capability to invoke the Plot Viewer.
5. ~~T~~The FD Details Viewer will provide a capability to refresh the [display with updated information](#). ~~FD Details Viewer if updated information becomes available.~~
6. ~~T~~The FD Details Viewer will provide a capability to expand itself to include the Data Fusion Viewer, the Data Health Viewer, the Change Coefficients Viewer, the PCL Viewer [and Constraint Viewer](#). If no ~~i~~information is available for a Viewer, it will be grayed out.
7. ~~T~~The FD Details Viewer will also *execute in the BASIS environment*.
8. [The FD Details Viewer will provide a capability to save print information to a file.](#)
9. ~~—~~
9. The FD Details Viewer will conform to the “CLCS HCI Style Guide and Standards” document number 84K00230.
10. The FD Details Viewer will provide a capability to show all information pertaining to an FD on one window.
11. The FD Details Viewer will provide a capability to print information on the screen to a hard copy device.

### 2.2.3 FD Details Viewer Performance Requirements

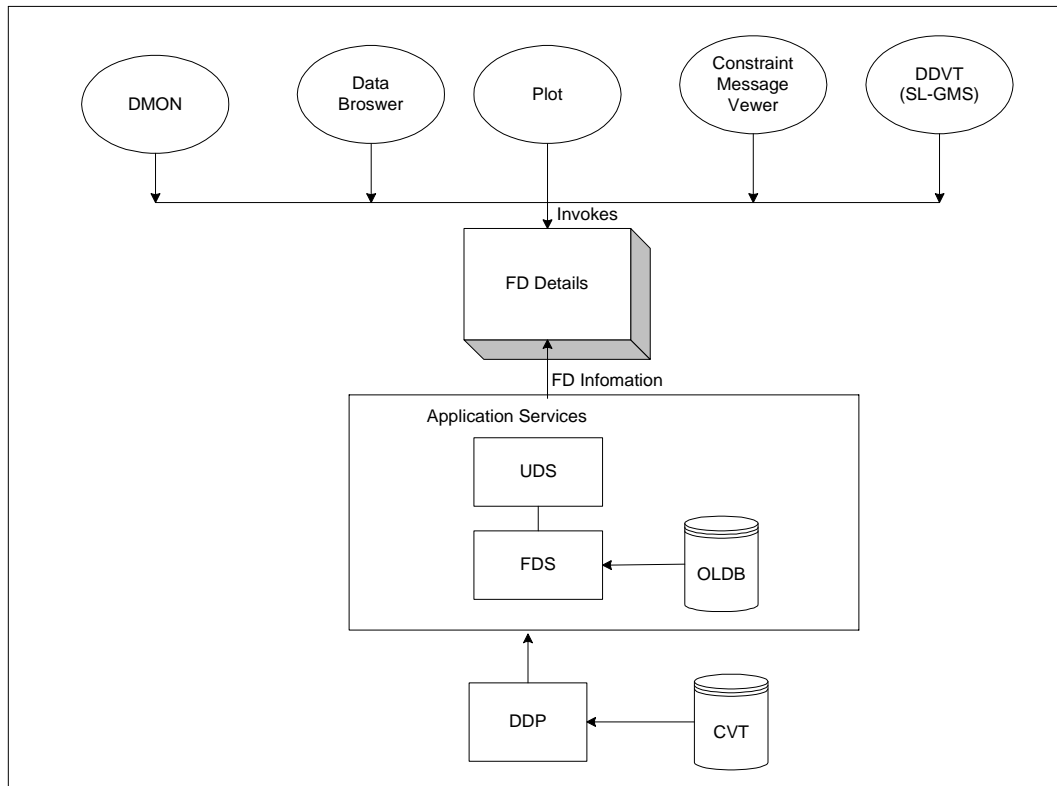
The FD Details Viewer will appear on the CCWS screen within two seconds of selection from the DDVT or Data Browser.

# Software Requirements and Design Specification

## 2.2.4 FD Details Viewer Interfaces

The FD Details Viewer will be invoked from either the DDVT , Data Browser, Constraint Details Viewer, Plot Viewer, or DMON Viewer.

## 2.2.5 FD Details Viewer Detailed Data Flow Diagram



The FD details Viewer requests and receives information from User Display Services.

## 2.3 FD Details Viewer Design Specification

The FD Details Viewer runs in the CCWS, and is invoked from the DDVT (SL-GMS Application) or from one of the following: the Data Browser, the Constraint Details Viewer, the Plot Viewer, or the DMON Viewer. It is implemented in Java. Each element of variable information on the FD Details Viewer screen is represented as either a text field or a drop down list box.

Appendix A contains a Class Diagram for the FD Details Viewer.

## 2.3.1 FD Details Viewer State Diagram

The FD Details Viewer State Diagram shown in Figure 4 relates the states and the events of the FD Details Viewer.

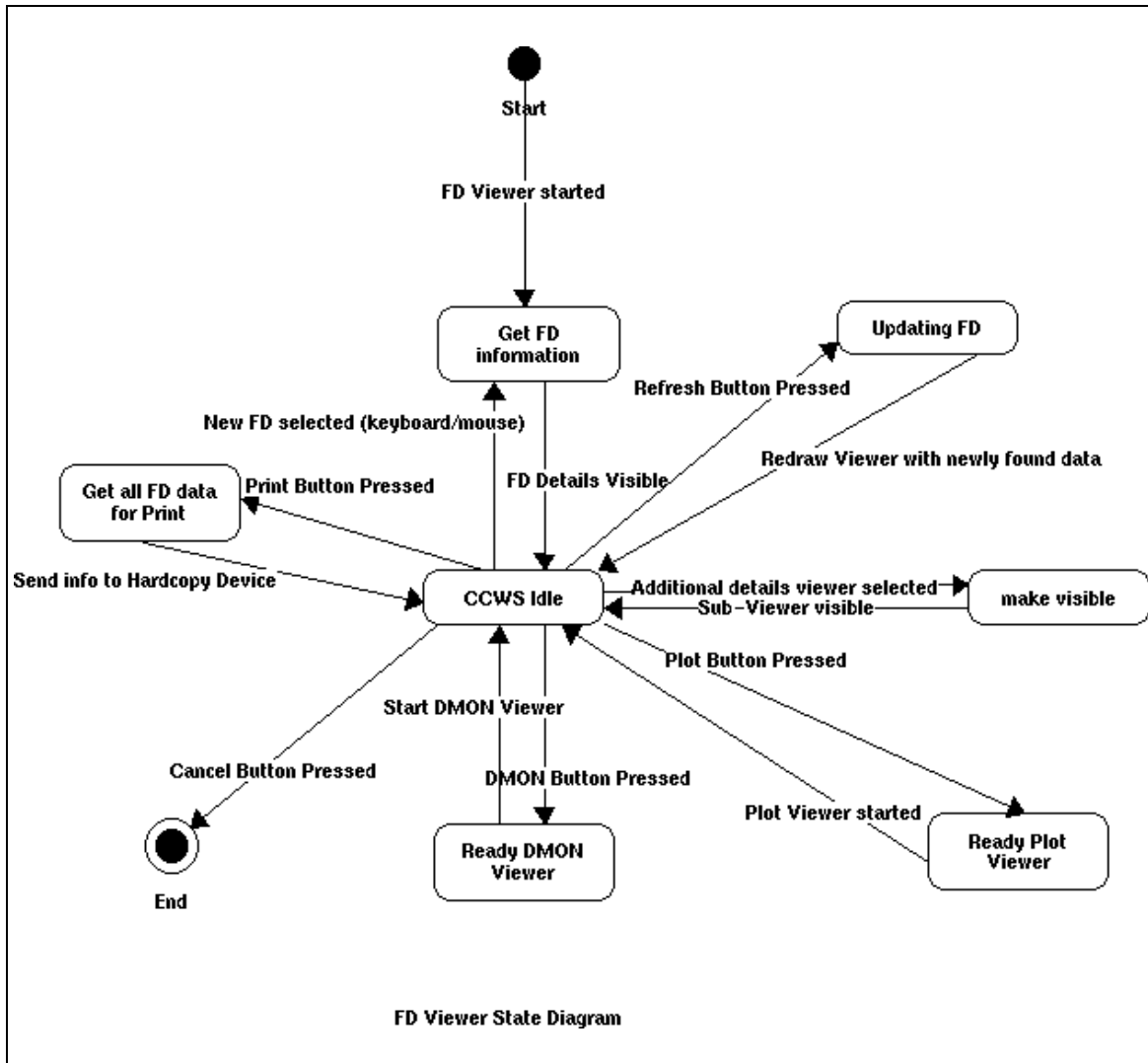


Figure 4. FD Details Viewer State Diagram.

## 2.3.2 FD Details Viewer External Interface

The FD Details Viewer will not provide any APIs. It will call the following API from the Data Handler within User Display Services. The Data Handler in UDS will send the updates to the FD Details Viewer. These interfaces are described in the User Display Services Interface Description Document (IDD) 84K00361.

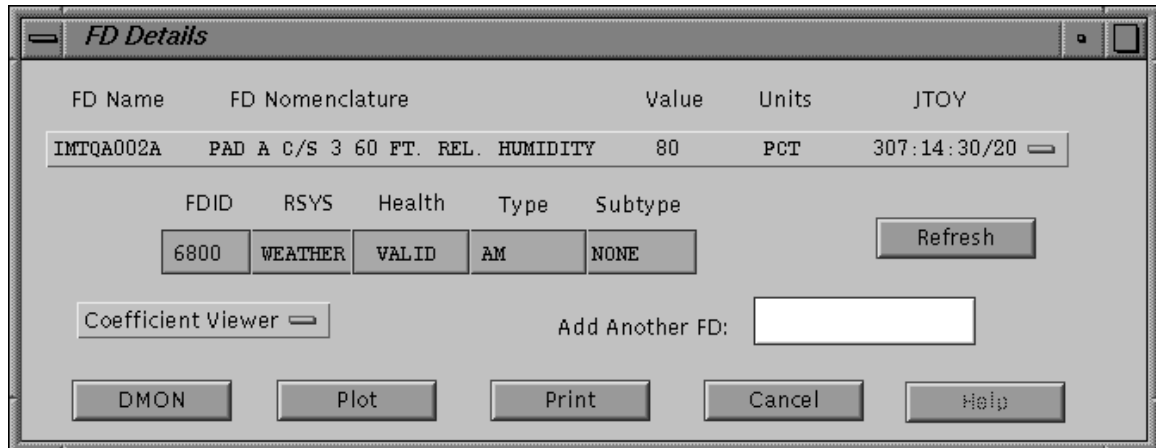
### 2.3.2.1 FD Details Viewer Message Formats

Not applicable.

# Software Requirements and Design Specification

## 2.3.2.2 FD Details Viewer Display Formats

The FD Details Viewer is invoked from the DDVT (SL-GMS Application) , Data Browser, Constraint Viewer, FD Details Viewer, Plot Viewer, or DMON Viewer.



**Figure 5. Main Window of FD Details Viewer without any Sub Viewers Displayed.**

### Main Window Functions

1. Display the FD Name, FD Nomenclature, Value, Units, Constraint, Time, FDID, RSYS, Health, Raw Counts, Type, Sub Type at the appropriate location listed in a scrolled window.
2. Allow users the capability to add another FD to the window and receive detailed information about it.
3. Allow users to refresh the window for updated data.
4. Allow users to invoke other Viewers from the window including DMON and Plot.
5. Allow users to invoke the following sub viewers: Coefficient, Constraint, Fusion, Health and PCL Viewers in the main window by using the viewer drop down list.
6. Allow the user to print the entire screen with the print button.
7. Allow the user to close the FD Details Window with a cancel button.
8. Allow the user to invoke the help screen with the help button.

## 2.3.2.3 FD Details Viewer Input Formats

The FD Details Viewer gets a list of FDs from the DDVT (SL-GMS Application) , Data Browser, Constraint Details Viewer, Plot Viewer, or DMON Viewer. An API call is made to User Display Services to provide the information for the list of FDs. These API interfaces are described in the User Display Services Interface Description Document (IDD) 84K00361.

# Software Requirements and Design Specification

## 2.3.2.4 FD Details Viewer Recorded Data

Not applicable.

## 2.3.2.5 FD Details Viewer Printer Formats

Not applicable. Only print screens will be accomplished with the FD Details Viewer.

## 2.3.2.6 FD Details Viewer Inter-process Communications

The FD Details Viewer receives data only from User Display Services (UDS) part of Application Services. The FD Details Viewer will display the FD Name, FD Nomenclature, Value, Units, Constraint, Time, FDID, RSYS, Health, Type and Sub Type.

For the other viewers including Constraint Details, Health, Fusion, Coefficient and PCL FD Details will handle all inter-process communications for data retrieval from UDS.

## 2.3.2.7 FD Details Viewer External Interface Calls

These interfaces are described in the User Display Services Interface Description Document (IDD) 84K00361.

## 2.3.3 FD Details Viewer Test Plan

The Data Browser system level tests may be run in either the IDE or SDE, or both. These tests are run on the CCWS platform. Application services will be required to send data to the Data Browser for FD Name, FD Nomenclature, Value, Units, Constraint, JTOY, FDID, RSYS, Health, Type and Sub Type.

The specific test cases that will be run include:

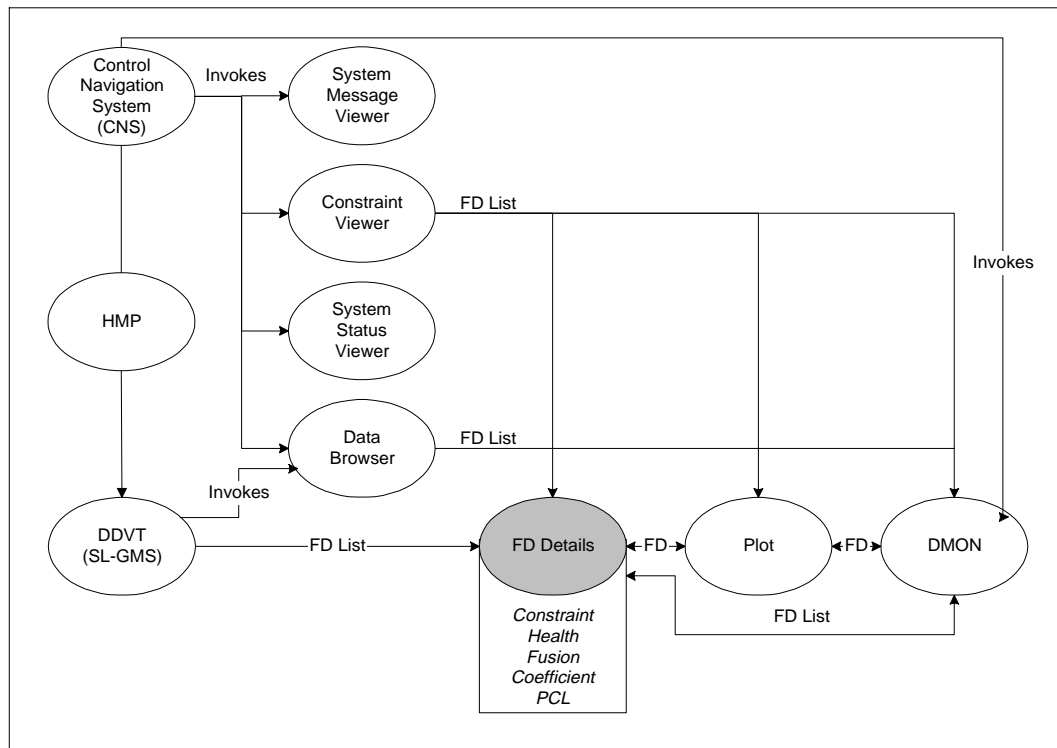
1. Verify that a FD Name, FD Nomenclature, Value, Units, Constraint, JTOY, FDID, RSYS, Health, Type, Sub Type is received in the FD Details Window.
2. Verify that the FD name drop down list is present when more than one FD is added to the FD Details Window.
3. Verify print screen capability.
4. Verify the DMON Button invokes the DMON Viewer.
5. Verify the Plot Button invokes the Plot View.
6. Verify the FD Details Button invokes the FD Details Window.
7. Verify the Cancel Button closes the Data Browser.
8. Verify the viewer drop down list pops up when left mouse clicking over the list.
9. Verify the viewer drop down list invokes the Coefficient Viewer.
10. Verify the viewer drop down list invokes the Constraint Viewer.
11. Verify the viewer drop down list invokes the Fusion Viewer.
12. Verify the viewer drop down list invokes the Health Viewer.
13. Verify the viewer drop down list invokes the PCL Viewer.
14. Verify the Add Another FD text input box adds another FD to the FD name drop down list.
15. *Verify the Help Button invokes the Help Window.*

## 3. Data Fusion Viewer

### 3.1 Data Fusion Viewer Introduction

#### 3.1.1 Data Fusion Viewer Overview

The Data Fusion Viewer is a part of the FD Details Viewer. The Data Fusion Viewer provides the Checkout and Launch Control System (CLCS) the ability to display detailed information pertaining to a fusion-based Function Designator (FD) to the Command and Control Workstation (CCWS). The Data Fusion Viewer displays details about a fused FD, its component FDs and the algorithm used to calculate the fused FD.



#### 3.1.2 Data Fusion Viewer Operational Description

The Data Fusion Viewer is invoked only from the FD Details Viewer.

The Data Fusion Viewer will display information about the fused FD that was selected on the FD Details Viewer. Only one fused FD is displayed on the Data Fusion Viewer at a time. Information is displayed regarding the component FDs that make up the fused FD. *Double-clicking on a component FD, or selecting a component FD then pressing the FD Details button, will cause an FD Details Viewer to be displayed for this component FD.*

Fused FD information displayed on this viewer consists of the FD nomenclature, fused FD value, time, health, component FD information, and algorithm information.

Component information displayed for this fused FD consists of the component FD nomenclature, value, time, health, and range.

# Software Requirements and Design Specification

The algorithm displayed on this viewer is the textual format of the algorithm used to compute the fused FD from the component FDs. A descriptive explanation of the algorithm is also displayed which explains the content of the algorithm.

## 3.2 Data Fusion Viewer Specifications

### 3.2.1 Data Fusion Viewer Ground Rules

1. An extracted descriptive text, consisting of a fused FD, component FDs, a textually formatted algorithm, and an algorithm descriptive text for the Data Fusion Viewer, is stored in an ASCII formatted file retrieved via Application Services.
2. The Data Fusion Viewer requests information pertaining to fused FDs through User Display Services and FD Services.
3. The Data Fusion Viewer is invoked from the FD Details Viewer.
4. In order to run on the BASIS environment, the On-Line Data Bank and System Configurations Table must be available. The BASIS environment will provide a monitoring only capability.

### 3.2.2 Data Fusion Viewer Functional Requirements

1. The Data Fusion Viewer will provide a window to display the components of a fused FD on the CCWS.
2. The Data Fusion Viewer will display low-level and high-level end-item state for component FDs. These boundary levels for end-item states show the range of values in analog inputs.
3. *The Data Fusion Viewer will provide a method to access the component FDs of a fused FD and invoke an FD Details Viewer to view detailed information about a component FD.*
4. The Data Fusion Viewer will display the value, health and time for the fused FD and the component FDs.
5. The Data Fusion Viewer will display the Fused FD algorithm.
6. The Data Fusion Viewer will display the Fused FD algorithm description.
7. The Data Fusion Viewer will conform to the "CLCS HCI Style Guide and Standards" documentation number 84K00230.
8. *The Data Fusion viewer will also execute in the BASIS environment.*
9. *The Data Fusion viewer will retrieve Fused FD and component FD information from the SDC if the CVT information is out of synchronization.*

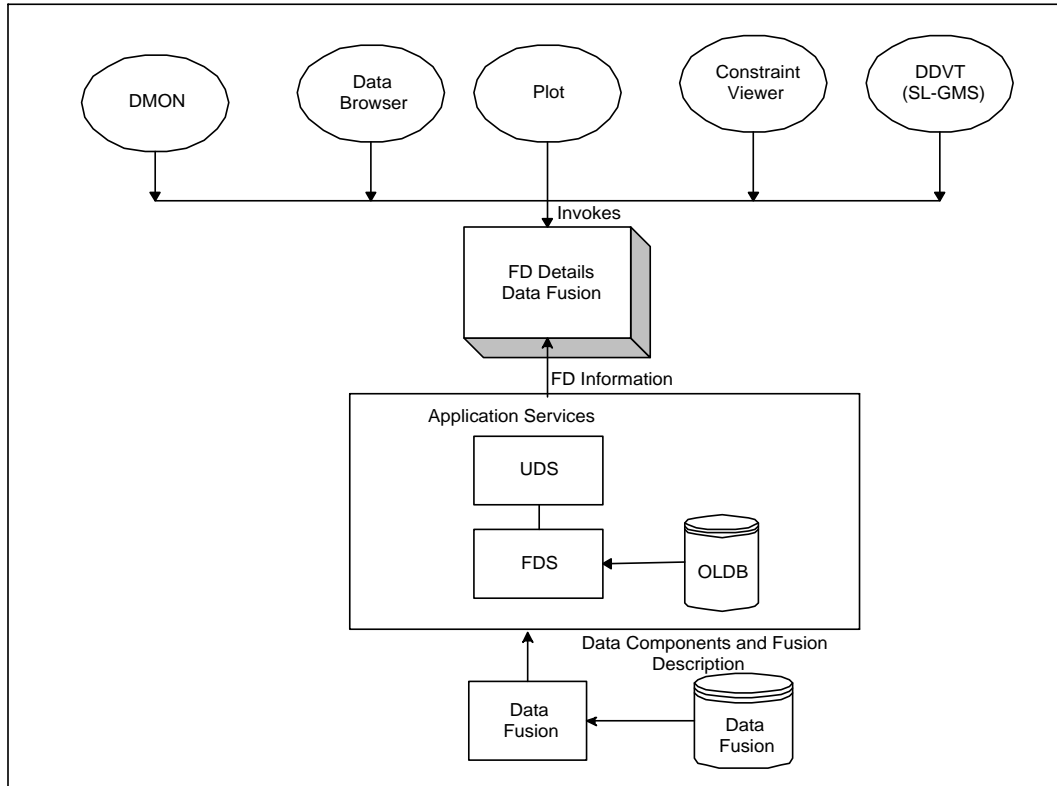
### 3.2.3 Data Fusion Viewer Performance Requirements

1. The Data Fusion Viewer will be displayed within two seconds from selecting the Data Fusion Viewer from the FD Details Viewer window.

### 3.2.4 Data Fusion Viewer Interfaces

The Data Fusion Viewer is invoked from the FD Details Viewer. The Data Fusion Viewer interfaces with Application Services to retrieve fused FD and component FD information.

## 3.2.5 Data Fusion Viewer Data Flow



## 3.3 Data Fusion Viewer Design Specification

The Data Fusion Viewer runs in the CCWS, and is invoked from the FD Details Viewer. It is implemented in Java. Each element of variable information on the Data Fusion Viewer screen is represented as either a Java text field, text area, or list box object.

The Data Fusion Viewer displays a “time homogeneous” snapshot, provided by User Display Services, of the required information. This assures that the component FDs and the algorithm produce the fused FD as displayed.

The Data Fusion Viewer Class Diagram is contained in the FD Viewer Class Diagram in Appendix A.

### 3.3.1 Data Fusion Viewer State Diagram

The Data Fusion Viewer displays the fusion data retrieved by the FD Details Viewer. *Double-Clicking on a component FD will kick-off a new FD Details Viewer for this component FD.*

The states and events of the Data Fusion Viewer are depicted in the FD Details Viewer State Diagram in the FD Details Viewer Design Specification.

### 3.3.2 Data Fusion Viewer External Interfaces

#### 3.3.2.1 Data Fusion Viewer Message Formats

Not Applicable.



## Software Requirements and Design Specification

### 3.3.2.2 Data Fusion Viewer Display Formats

The Data Fusion Viewer is invoked from the FD Details Viewer. The FD Details Viewer expands to display the fused FD information, including the component FDs of the fused FD, and the Algorithm and Algorithm Description.

| FD Name   | FD Nomenclature                  | Value | Units | JTOY         |
|-----------|----------------------------------|-------|-------|--------------|
| IMTQA002A | PAD A C/S 3 60 FT. REL. HUMIDITY | 80    | PCT   | 307:14:30/20 |

| FDID | RSYS | Health  | Type | Subtype |
|------|------|---------|------|---------|
| 6879 | HMF  | INVALID | NONE | NONE    |

| Component FD | FD Nomenclature        | Value | JTOY        | Health | Analog Low | Analog High |
|--------------|------------------------|-------|-------------|--------|------------|-------------|
| H42X1120X    | OX HE ISO VLV A OP IND | OPEN  | 307:1430/20 | Valid  |            |             |
| H42X1121X    | OX HE ISO VLV A CL IND | OPEN  | 307:1430/20 | Valid  |            |             |

Algorithm

LV102STATE = H42X1120X .AND. H42X1121X

Algorithm Description

LV102STATE IS THE FUSED FD FOR H42X1120X AND H42X1121X

Figure 3: Data Fusion Viewer (as pulled down from FD Details Viewer.)

Data Fusion Viewer Functions:

1. For the Component FDs, displays the FD Name, FD Nomenclature, Value, JTOY, Health, Analog Low limit and the Analog High limit, the Algorithm, and the Algorithm Description.
2. Provides the capability to invoke a new FD Details Viewer for a selected Component FD, by either double-clicking on the Component FD or clicking once on the Component FD in order to highlight it, then pressing the FD Details button.
3. Provides the capability to close the Data Fusion Viewer with a cancel button.

## 3.3.2.3 Data Fusion Viewer Input Formats

Not Applicable.

## 3.3.2.4 Data Fusion Viewer Recorded Data

Not Applicable.

## 3.3.2.5 Data Fusion Viewer Printer Formats

Not Applicable.

## 3.3.2.6 Data Fusion Viewer Inter-process Communications

Not Applicable.

## 3.3.2.7 Data Fusion Viewer External Interface Calls

The Data Fusion Viewer has no external interface calls. The interface calls that are required to retrieve Data Fusion Viewer data are made by the FD Details Viewer. These interfaces are described in the User Display Services Interface Description Document (IDD) 84K00361.

## 3.3.2.8 Data Fusion Viewer Table Formats

Not applicable.

## 3.3.3 Data Fusion Viewer Test Plan

The Data Fusion system level tests may be run in either the IDE or SDE, or both. These tests are run on the CCWS platform.

The specific test cases that will be run include:

1. Verify that the Data Fusion Viewer is displayed on the CCWS.
2. Verify that the data displayed in the component list area on the Data Fusion Viewer includes:
  - Component FD
  - FD Nomenclature
  - Value
  - JTOY
  - Health
  - Low Limit (of Range)
  - High Limit (of Range)
3. Verify that the slide bars on the Data Fusion Viewer component list area can be moved in order to scroll through all the data that is contained in this area.
4. Verify that the Data Fusion Algorithm for the Fused FD displays on the Data Fusion Viewer.
5. Verify that the slide bars on the Data Fusion Algorithm field can be moved in order to scroll through all the data that is contained in the field.
6. Verify that the Data Fusion Algorithm Description for the Fused FD displays on the Data Fusion Viewer.
7. Verify that the slide bars on the Data Fusion Algorithm Description field can be moved in order to scroll through all of the data that is contained in this field.

## Software Requirements and Design Specification

8. *Verify that the FD Details Viewer can be displayed for a Component FD by double-clicking (selecting) on a Component FD and verifying that the FD Details Viewer is displayed for the selected Component FD.*
9. *Verify that the FD Details Viewer can be displayed for a Component FD by high-lighting (selecting) a Component FD by clicking once on the Component FD, pressing the FD Details button, and verifying that the FD Details Viewer is displayed for the selected Component FD.*
10. Verify that the Data Fusion Viewer is no longer displayed and the FD Details Viewer is resized to exclude the Data Fusion Viewer when the Cancel button is pressed.

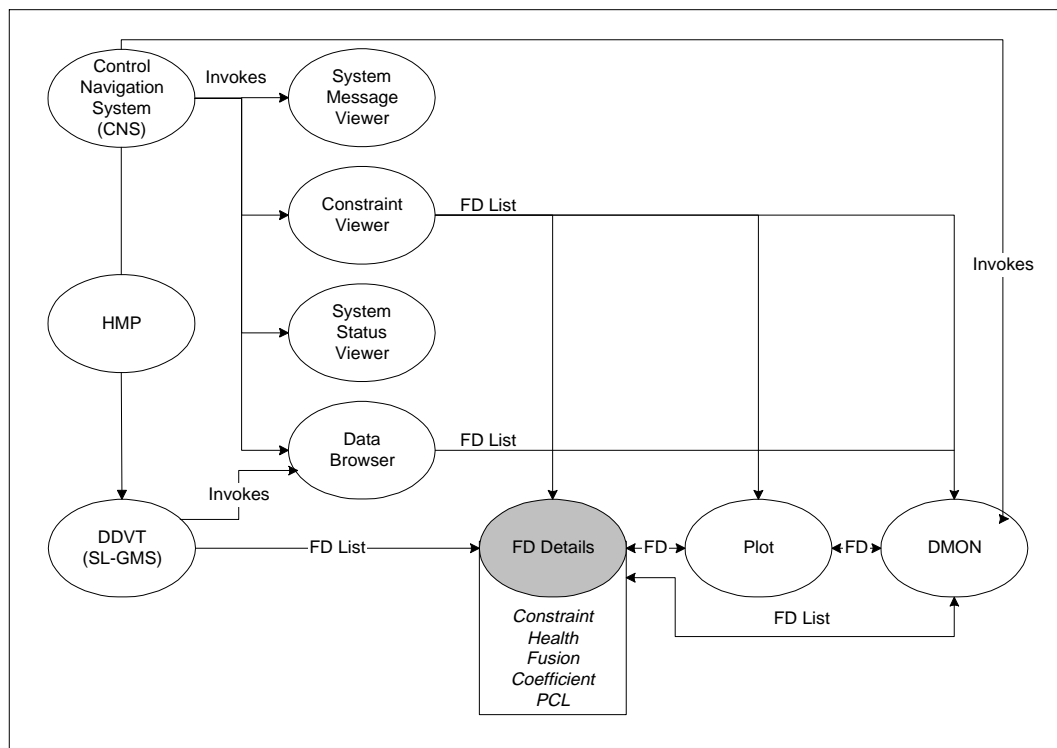
## 4. Data Health Viewer

### 4.1 Data Health Viewer Introduction

#### 4.1.1 Data Health Viewer Overview

The Data Health Viewer provides the Checkout and Launch Control Data System (CLCS) with a display to view the health and status of a Function Designator (FD) or set of FDs on a CCWS. The Data Health Viewer is part of the enlarged FD Details Viewer.

The diagram below shows the hierarchy of the System Viewers.



#### 4.1.2 Data Health Viewer Operational Description

The Data Health Viewer is invoked from the FD Details Viewer. The Data Health Viewer will elaborate on the “Reason” description of the problem. This description will provide a detailed explanation of the problem

## 4.2 Data Health Viewer Specifications

### 4.2.1 Data Health Viewer Ground Rules

1. Data health information will be received from a User Display Services API(s).
2. In order to run on the BASIS environment the On-Line Data Bank and System Configuration Table must be available. The BASIS environment will provide monitoring only capability.

### 4.2.2 Data Health Viewer Functional Requirements

1. The Data Health Viewer will provide a window to display Data Health for an FD on the CCWS.
2. The Data Health Viewer will provide a color designation indicating the validity of the data viewed.
3. The Data Health Viewer will provide the ability to examine reason code and text.
4. The Data Health Viewer will conform to the "CLCS HCI Style Guide and Standards" document number 884K00230.
5. *The Data Health Viewer will also execute in the BASIS environment.*
6. The Data Health Viewer will provide the capability to print a snapshot of the Data Health Viewer window.

### 4.2.3 Data Health Viewer Performance Requirements

- The Data Health Viewer will appear on the screen within two seconds of invocation.

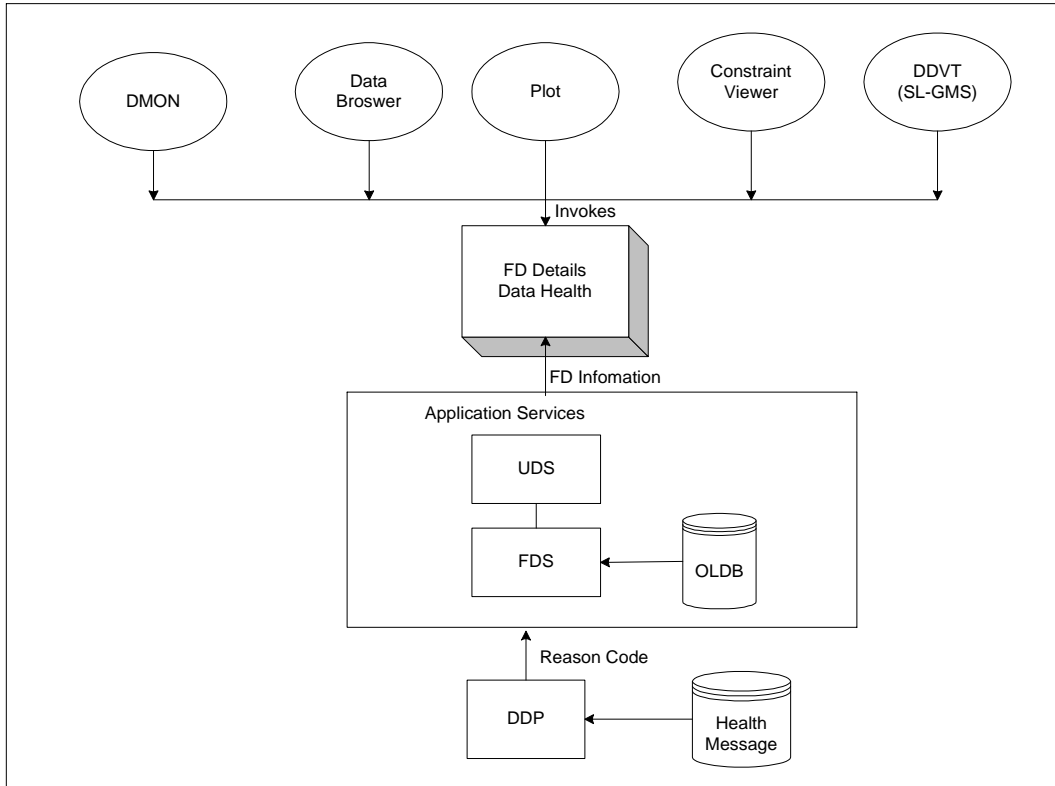
### 4.2.4 Data Health Viewer Interfaces

The Data Health Viewer interfaces with FD Services to get data health information from the CVT, OLDB, and the Data Health Catalog.

# Software Requirements and Design Specification

## 4.2.5 Data Health Viewer Data Flow Diagram

The FD Details Viewer will receive information from User Display Services.



# Software Requirements and Design Specification

## 4.3 Data Health Viewer Design Specification

The Data Health Viewer runs on the CCWS platform and is invoked from the FD Details Viewer. It is implemented in Java. The Data Health Viewer displays the data health status for an FD selected in FD details. For an invalid health status, the Data Health Viewer will obtain a reason code number from the CVT using an API call. The reason code text string is then looked up using the Reason Code number through an API call to Application Services. Reason code messages will be displayed, along with the time of the request, in a scrollable list. The Data Health Viewer captures a “time homogeneous” snapshot of the required information. This assures that the data health bits are consistent with the value and time represented.

The Data Health Viewer Class Diagram is contained in the FD Viewer Class Diagram in Appendix A.

### 4.3.1 Data Health Viewer State Diagram

The states and events of the Data Health Viewer are depicted in the FD Details Viewer State Diagram in the FD Details Viewer Design Specification.

### 4.3.2 Data Health Viewer External Interfaces

The Data Health Viewer has no external interface calls. The interface calls that are required to retrieve are made by the FD Details Viewer. These interfaces are described in the User Display Services Interface Description Document (IDD) 84K00361.

#### 4.3.2.1 Data Health Viewer Message Formats

Not applicable.

## 4.3.2.2 Data Health Viewer Display Formats

The Data Health Viewer is invoked from the FD Details Viewer drop down list box.

The screenshot shows a window titled "FD Details" with a table of data and several control buttons. The table has columns for FD Name, FD Nomenclature, Value, Units, and JTOY. The data row shows "IMTQA003A", "PAD A C/S 3 30 FT. REL. HUMIDITY", "79", "PCT", and "307:14:30/20". Below the table, there are buttons for "Health Viewer", "DMON", "Plot", "Print", "Cancel", and "Help". A "Refresh" button is also present. A section for "Add Another FD:" includes a text input field. At the bottom, there is a section for "FD Name: IMTQA003A" with a "Valid" button, a "Refresh" button, and a "Cancel Health" button. A large text area displays "Valid" and "Reason # ReasonCode JTOY".

| FD Name   | FD Nomenclature                  | Value | Units | JTOY         |
|-----------|----------------------------------|-------|-------|--------------|
| IMTQA003A | PAD A C/S 3 30 FT. REL. HUMIDITY | 79    | PCT   | 307:14:30/20 |

FDID: 3386 RSYS: WEATHER Health: VALID Type: AM Subtype: NONE

Buttons: Refresh, Health Viewer, DMON, Plot, Print, Cancel, Help

Add Another FD:

FD Name: IMTQA003A Reason # ReasonCode JTOY

Buttons: Valid, Refresh, Cancel Health

Figure 4. Data Health Viewer Window with Valid Data Health Status



## Software Requirements and Design Specification

The screenshot shows a window titled "FD Details". It contains a table with the following data:

| FD Name   | FD Nomenclature                  | Value | Units | JTOY         |
|-----------|----------------------------------|-------|-------|--------------|
| IMTQA003A | PAD A C/S 3 30 FT. REL. HUMIDITY | 79    | PCT   | 307:14:30/20 |

Below the table, there are several buttons and a "Refresh" button. The buttons are labeled "FDID", "RSYS", "Health", "Type", and "Subtype". The values for these buttons are "3386", "WEATHER", "VALID", "AM", and "NONE" respectively. There is also a "Refresh" button.

Below the buttons, there is a "Health Viewer" button and an "Add Another FD:" label with a text input field.

At the bottom, there are buttons labeled "DMON", "Plot", "Print", "Cancel", and "Help".

Below the buttons, there is a section for "FD Name: IMTQA003A". It shows "Reason # 257" and "ReasonCode GSE Gateway not communicating". The "JTOY" value is "307:14:30/20". There is an "Invalid" button and a "Refresh" button. At the bottom of this section is a "Cancel Health" button.

**Figure 5. Data Health Viewer Window with Invalid Data Health Status**

The Data Health Viewer's functions are:

1. Display the FD name.
2. Display the FD data health status as valid or invalid.
3. Display, for invalid data health, the reason code number, reason code text and the time of the data snapshot.
4. The refresh button, when pressed, will cause the FD data health status to be updated.
5. The Cancel button, when pressed, will close the Data Health Viewer window.

### 4.3.2.3 Data Health Viewer Input Formats

Not applicable.

### 4.3.2.4 Data Health Viewer Recorded Data

Not applicable.

### 4.3.2.5 Data Health Viewer Printer Formats

Not applicable. Only print screens will be available with the Data Health Viewer.

### 4.3.2.6 Data Health Viewer Inter-process Communications

The Data Health Viewer receives data only from Applications Services.

The Data Health Viewer will display FD name, data health status, reason code number, reason code text and JTOY.

## 4.3.2.7 Data Health Viewer External Interface Calls

The Data Health Viewer will not provide any APIs. It will call the following API from the Data Handler within User Display Services. The Data Handler in UDS will send the updates to the Data Health Viewer. These interfaces are described in the User Display Services Interface Description Document (IDD) 84K00361.

## 4.3.3 Data Health Viewer Test Plan

The Data Health Viewer will run on either the IDE or the SDE or both.

These test are run on the CCWS platform.

Application services will be required to send data to the Data Health Viewer for FD Health.

The specific test cases that will be run include:

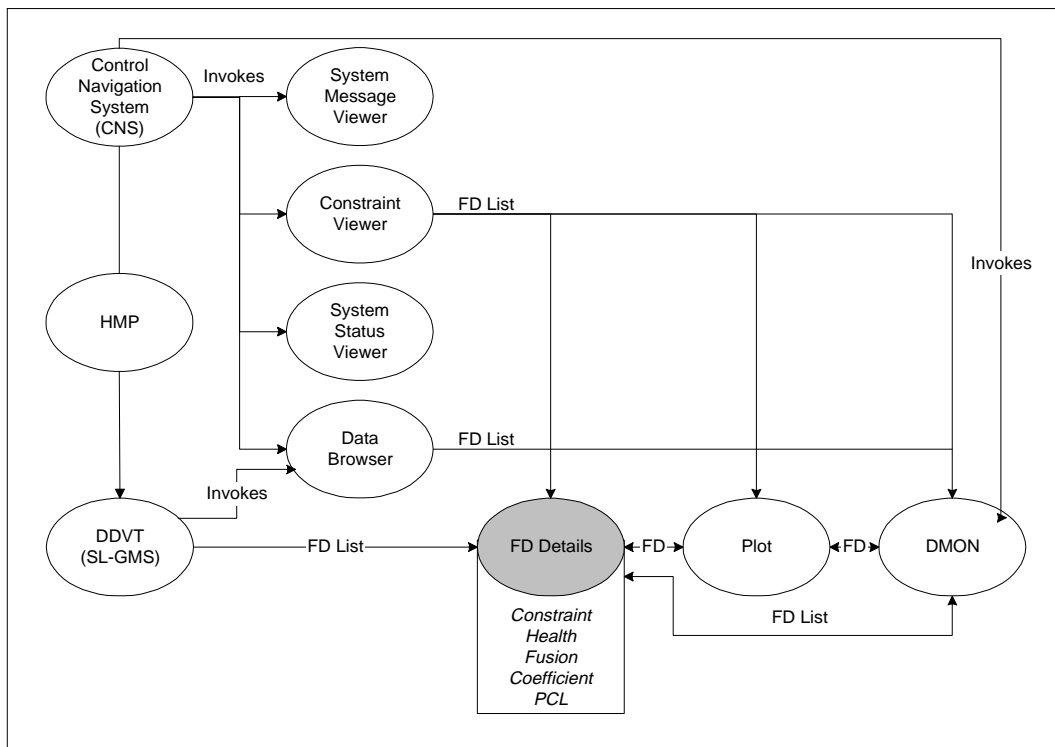
1. Verify that the Data Health Viewer is displayed with the correct FD name when it is invoked.
2. Verify that the correct Data Health status is displayed for the selected FD.
3. Verify that the correct Reason Code is displayed for an Invalid health status.
4. Verify that the correct Reason Code text is displayed for an Invalid health status.
5. Verify that the correct time is displayed.
6. Verify that when the Refresh button is pushed, the health status is updated.
7. Verify that when the Print button is pushed, the print function is invoked.
8. Verify that when the Cancel button is pushed, the DHV display is closed.
9. Verify that when the Help button is pushed, the Help function is invoked.

## 5. PCL Viewer

### 5.1 PCL Viewer Introduction

#### 5.1.1 PCL Viewer Overview

The PCL Viewer provides the Checkout and Launch Control System (CLCS) the ability to display on the Command and Control Workstation (CCWS) detailed information pertaining to a Prerequisite Control Logic Function Designator (FD). The PCL Viewer is part of the FD Details Viewer. The PCL Viewer displays details about a PCL.



#### 5.1.2 PCL Viewer Operational Description

The PCL Viewer is expanded from the FD Details Viewer. The PCL Viewer will display information about the FDs associated with Prerequisite Control Logic.

## 5.2 PCL Viewer Specifications

### 5.2.1 PCL Viewer Ground Rules

1. Application Services will provide the API(s) to return the PCL Description.
2. In order to run on the BASIS environment the On-Line Data Bank and System Configuration Table must be available. The BASIS environment will provide monitoring only capability.

# Software Requirements and Design Specification

## 5.2.2 PCL Viewer Functional Requirements

1. The PCL Viewer will provide a window to display the list of FDs associated with Prerequisite Control Logic.
2. The PCL Viewer will display the PCL description for each FD.
3. The PCL Viewer will display the FD corresponding to the displayed FD in the FD Details viewer at the top of the PCL Viewer.
4. The PCL Viewer will provide the capability to sort the display results based on the FD name, Nomenclature, or PCL description.
5. The PCL Viewer will conform to the “CLCS HCI Style Guide and Standards” document number 84K00230.
6. *The PCL Viewer will also execute in the BASIS environment.*

## 5.2.3 PCL Viewer Performance Requirements

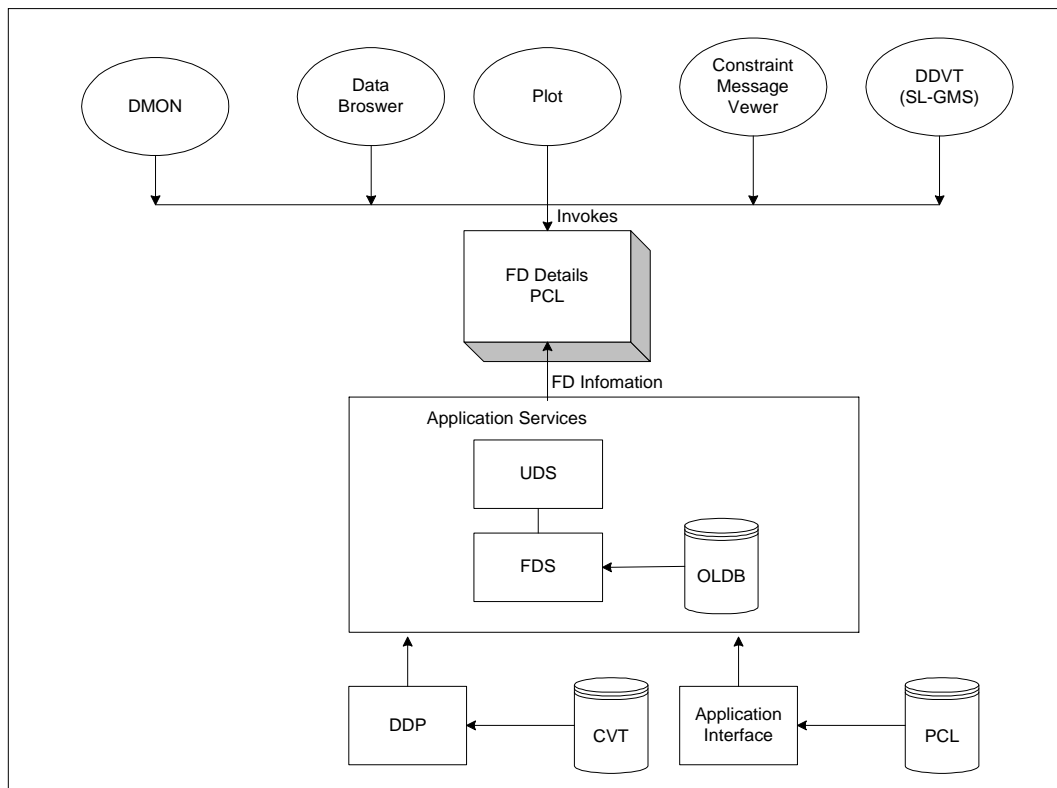
1. The PCL Viewer will be displayed within two seconds from selecting the PCL Viewer from the FD Details viewer window.

## 5.2.4 PCL Viewer Interfaces

The PCL Viewer is expanded from the FD Details Viewer.

## 5.2.5 PCL Viewer Data Flow

PCL Viewer will receive information from User Display Services.



# Software Requirements and Design Specification

## 5.3 PCL Viewer Design Specification

The PCL Viewer runs in the CCWS, and is invoked from the FD Details Viewer. It is implemented in Java. Each element of variable information on the PCL Viewer screen is represented as a text field.

The PCL Class Diagram is contained in the FD Viewer Class Diagram in Appendix A.

### 5.3.1 PCL Viewer State Diagram

The states and events of the PCL Viewer are depicted in the FD Details Viewer State Diagram in the FD Details Viewer Design Specification.

### 5.3.2 PCL Viewer External Interface

Not applicable.

#### 5.3.2.1 PCL Viewer Message Formats

Not applicable.

#### 5.3.2.2 PCL Viewer Display Formats

The PCL Viewer is invoked via the DDVT (SL-GMS Application) by way of the FD Details Viewer Window or CNS.

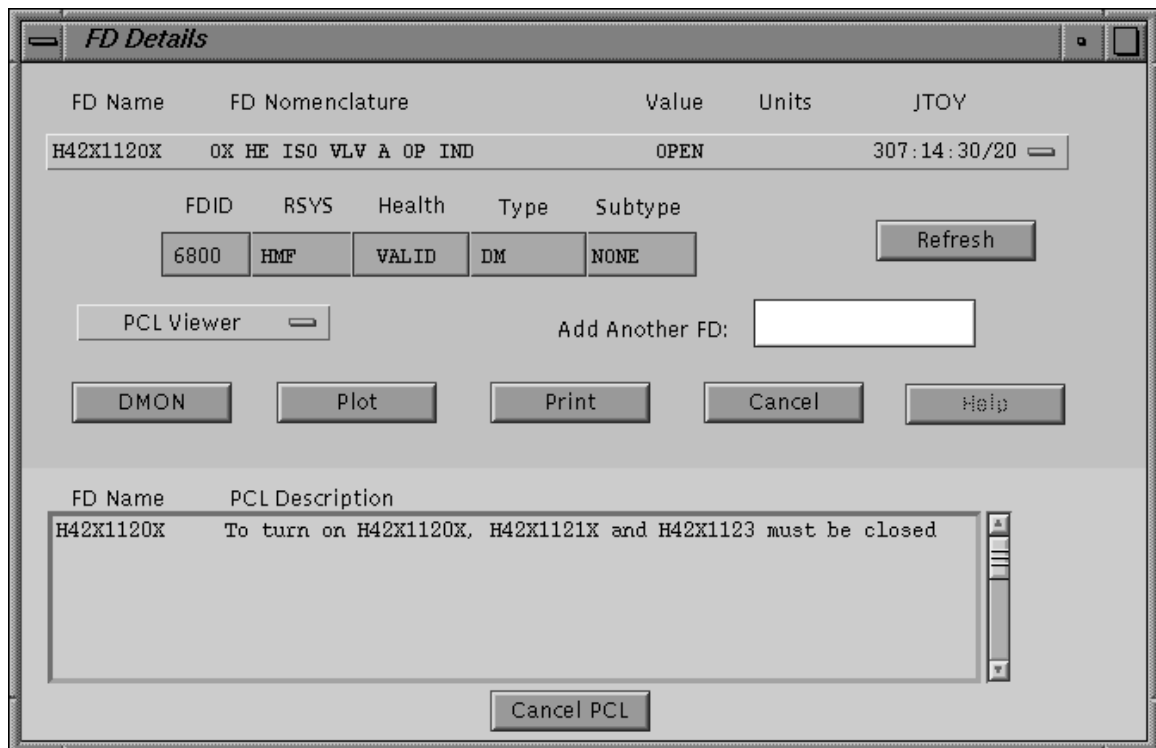


Figure 6. Main Window of PCL Viewer.

#### Main Window Functions

1. Display the FD Name and PCL Description at the appropriate location listed in a scrolled window.
2. Allow the user to close the PCL Viewer with a cancel button.

## **5.3.2.3 PCL Viewer Input Formats**

Not applicable.

## **5.3.2.4 PCL Viewer Recorded Data**

Not applicable.

## **5.3.2.5 PCL Viewer Printer Formats**

Not applicable.

## **5.3.2.6 PCL Viewer Inter-process Communications**

Not applicable.

## **5.3.2.7 PCL Viewer External Interface Calls**

The PCL Viewer has no external interface calls. The interface calls that are required to retrieve PCL Name and PCL Description are made by the FD Details Viewer. These interfaces are described in the User Display Services Interface Description Document (IDD) 84K00361.

## **5.3.3 PCL Viewer Test Plan**

The PCL Viewer system level tests may be run in either the IDE or SDE, or both. These tests are run on the CCWS platform. Application services will be required to send data to the PCL Viewer for FD Name and PCL Description.

The specific test cases that will be run include:

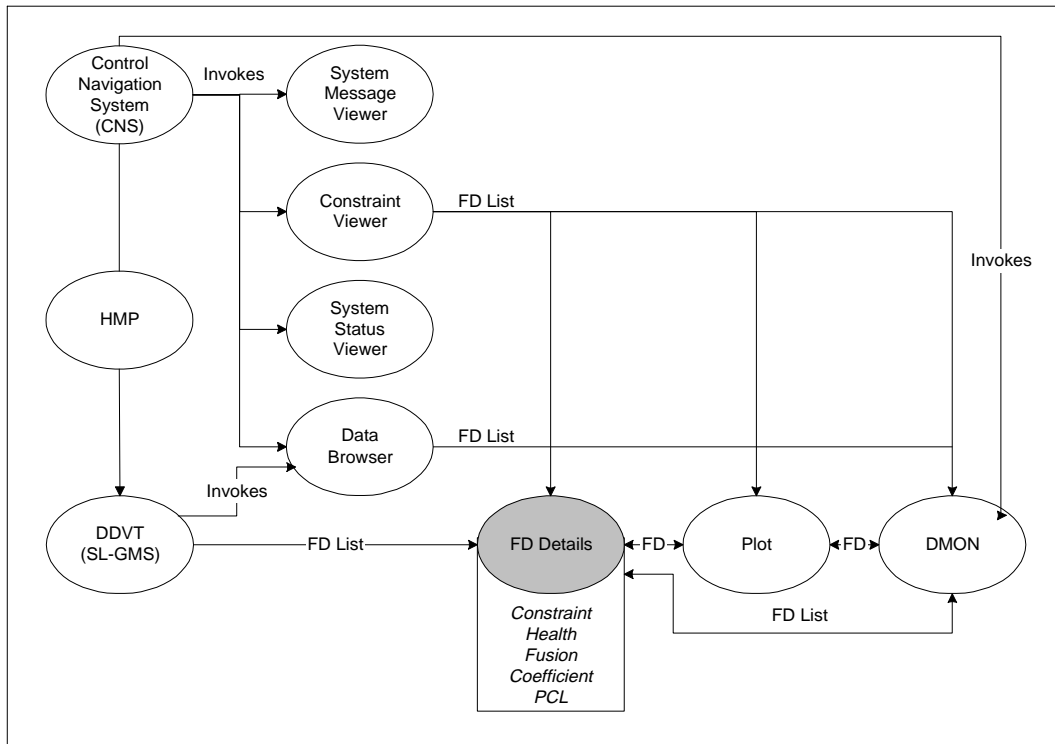
1. Verify a FD Name and PCL Description are received in the PCL Viewer.
2. Verify the Cancel Button closes the PCL Viewer.

## 6. Coefficient Viewer

### 6.1 Coefficient Viewer Introduction

#### 6.1.1 Coefficient Viewer Overview

The Coefficient Viewer provides the Checkout and Launch Control System (CLCS) the ability to display the coefficients on the Command and Control Workstation (CCWS) and allows the user to change these coefficients. The Coefficient Viewer is part of the FD Details Viewer.



#### 6.1.2 Coefficient Viewer Operational Description

The Coefficient Viewer is invoked from the FD Details Viewer.

The Coefficient Viewer will display the current coefficients and allow the user to change the values. “Apply” and “Cancel” buttons will allow the user to apply the changes or cancel the changes that have been entered on the viewer.

## 6.2 Coefficient Viewer Specifications

### 6.2.1 Coefficient Viewer Ground Rules

1. Application Services will provide an API(s) to return coefficient information.
2. Application Services will provide an API(s) to Coefficients.
3. Application Services will provide an API(s) to check Coefficient authentication.
4. Coefficient Viewer will not run in the BASIS environment. The BASIS environment will provide monitoring only capability.

### 6.2.2 Coefficient Viewer Functional Requirements

1. The Coefficient Viewer will provide a window to display calibration coefficients of an FD that are currently used by a gateway. The viewer will be displayed on the CCWS
2. The Coefficient Viewer will provide fields on the viewer for entry of the changed coefficient values.
3. The Coefficient Viewer will provide a “Apply” button, which upon pressing will apply the changes in the coefficient values that have been entered by the user.
4. The Coefficient Viewer will provide a “Cancel” button, which upon pressing will cancel any changes in the coefficient values that have been entered by the user.
5. The Coefficient Viewer will conform to the “CLCS HCI Style Guide and Standards” document number 84K00230.
6. *The Coefficient Viewer will have a capability to access the coefficients for an FD used at system build time.*

### 6.2.3 Coefficient Viewer Performance Requirements

1. The Coefficient Viewer will be displayed within two seconds from selecting the Coefficient Viewer from the FD Details Viewer.

### 6.2.4 Coefficient Viewer Interfaces

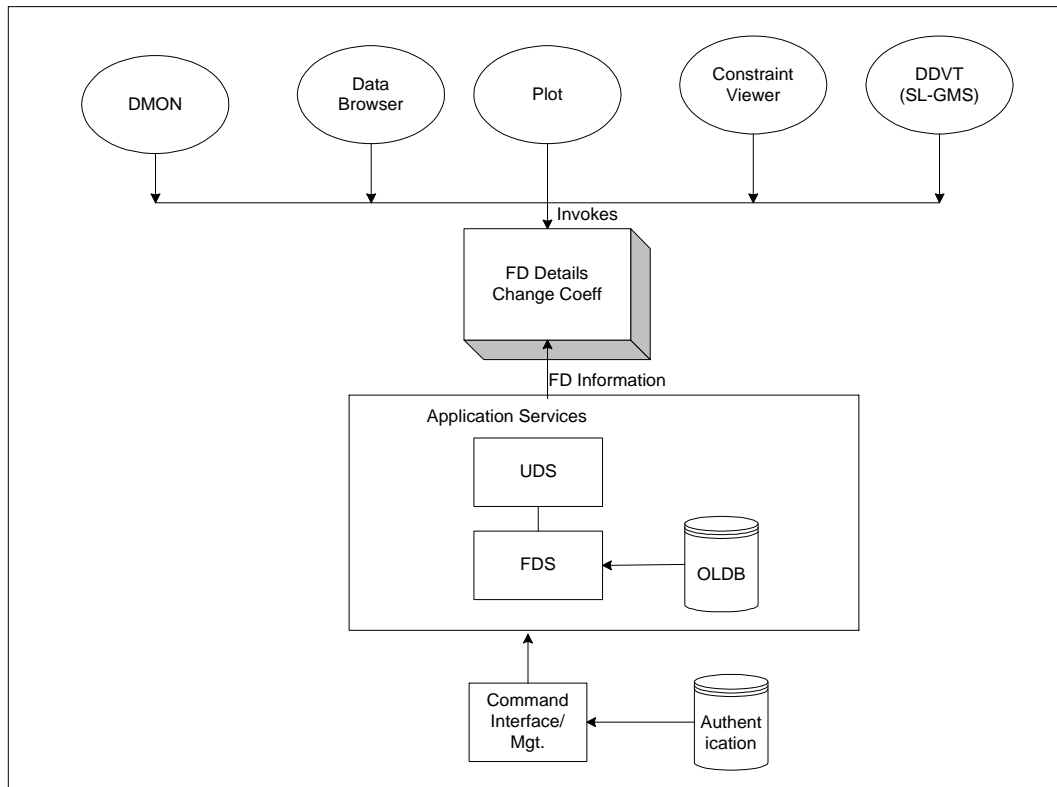
The Coefficient Viewer is invoked from the FD Details Viewer.



# Software Requirements and Design Specification

## 6.2.5 Coefficient Viewer Data Flow

The Coefficient Viewer will receive coefficient information from Application Services. Application Services will give the changed coefficients to command to check and update the curve coefficients.



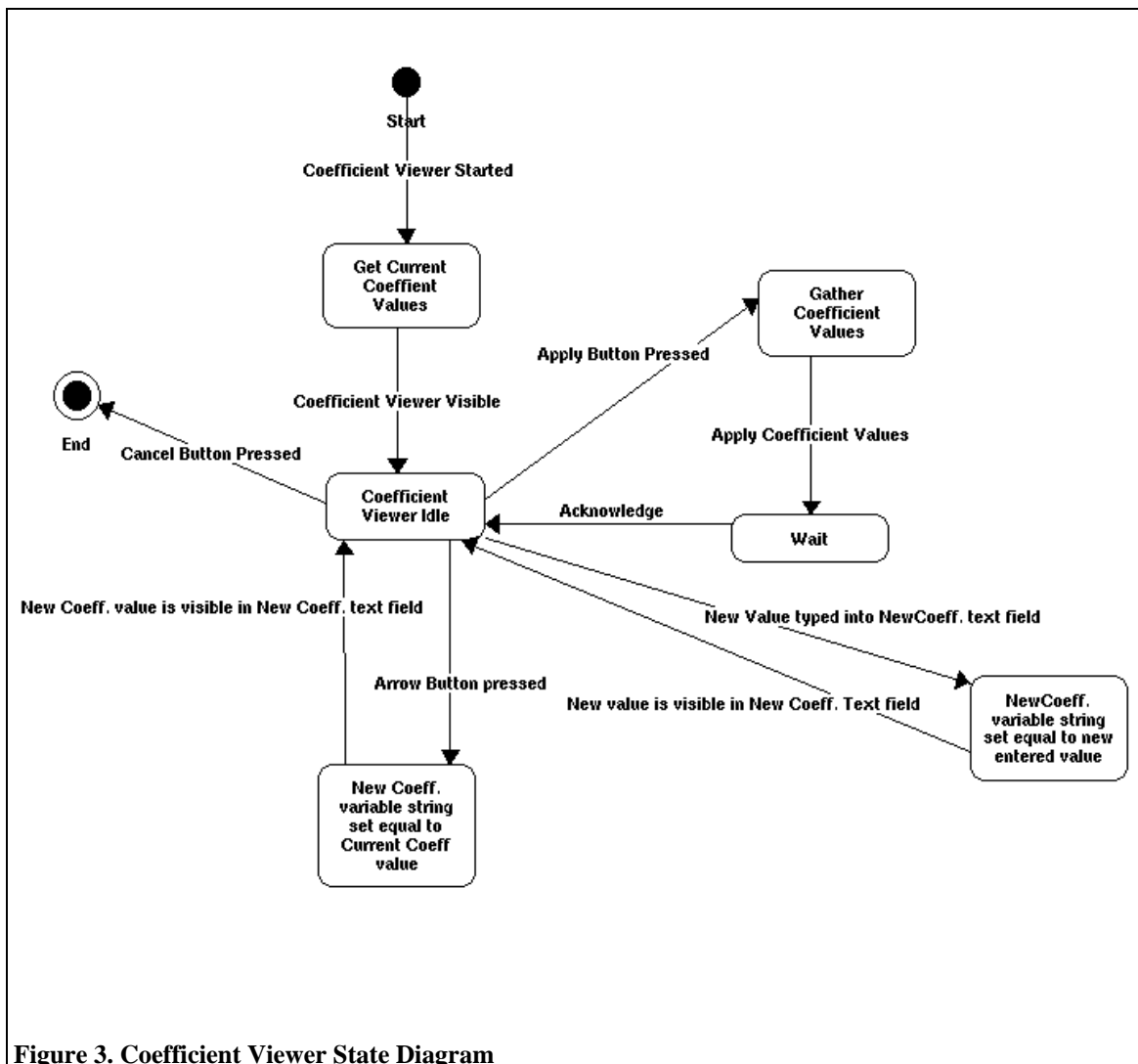
## 6.3 Coefficient Viewer Design Specification

The Coefficient Viewer runs on the CCWS platform and is invoked from the FD Details Viewer. It is implemented in Java. The Coefficient Viewer reads the current values of the calibration coefficients and displays them in text strings that cannot be edited. Text fields are provided to allow the user to type in up to six new values for the coefficients. An apply button is provided, which when pressed, will apply the changes via command to the gateway. Strict user permissions, by command management, will be checked before allowing any commanding. Once the changes are applied, the viewer will wait for acknowledgment that the changes were accepted.

The Coefficient Viewer Class Diagram is contained in the FD Viewer Class Diagram in Appendix A.

### 6.3.1 Coefficient Viewer State Diagram

The Coefficient Viewer State Diagram shown below relates the states and the events of the Coefficient Viewer.



## 6.3.2 Coefficient Viewer External Interfaces

The Data Health Viewer has no external interface calls. The interface calls that are required to retrieve are made by the FD Details Viewer. These interfaces are described in the User Display Services Interface Description Document (IDD) 84K00361.

### 6.3.2.1 Coefficient Viewer Message Formats

Not applicable.

### 6.3.2.2 Coefficient Viewer Display Formats

The Coefficient Viewer is invoked from the FD Details Viewer drop down list.

**Figure 4. Coefficient Viewer Window**

Coefficient Viewer Window Functions:

1. The current values of the calibration coefficients are displayed for up to six coefficient values.
2. If there are fewer than six coefficient values, the remaining will be grayed out and disabled.
3. Text fields are displayed to allow entry of up to six new calibration coefficient values.
4. The Apply button, when pressed, will apply the new coefficient values.
5. The Coefficient viewer, after applying the new values, will wait for acknowledgment confirming the command has been completed.

# Software Requirements and Design Specification

6. The Cancel button when pressed will close the Coefficient Viewer window.

## 6.3.2.3 Coefficient Viewer Input Formats

Not applicable.

## 6.3.2.4 Coefficient Viewer Recorded Data

Not applicable.

## 6.3.2.5 Coefficient Viewer Printer Formats

Not applicable.

## 6.3.2.6 Coefficient Viewer Inter-process Communications

The Coefficient Viewer receives data only from Applications Services.

The Coefficient Viewer will display the current values of the calibration coefficients and text fields to enter new coefficient values.

## 6.3.2.7 Coefficient Viewer External Interface Calls

The Coefficient Viewer will not provide any APIs. It will call an API from the Data Handler within User Display Services. The Data Handler in UDS will send the current values to the Coefficient Viewer. These interfaces are described in the User Display Services Interface Description Document (IDD) 84K00361.

## 6.3.3 Coefficient Viewer Test Plan

The Coefficient Viewer will run on either the IDE or the SDE or both.

These test are run on the CCWS platform.

The specific test cases that will be run include:

1. Verify that the Coefficient Viewer is displayed with the correct values for the current coefficients upon invoking the viewer.
2. Verify that the required user authentication is checked.
3. Verify that a new set of coefficients can be entered into the displayed fields.
4. Verify that the changed coefficients can be sent to and received by the gateway.
5. Verify that when the Print button is pushed, the print function is invoked.
6. Verify that when the Cancel button is pushed, the display is closed.
7. Verify that when the Help button is pushed, the Help function is invoked.

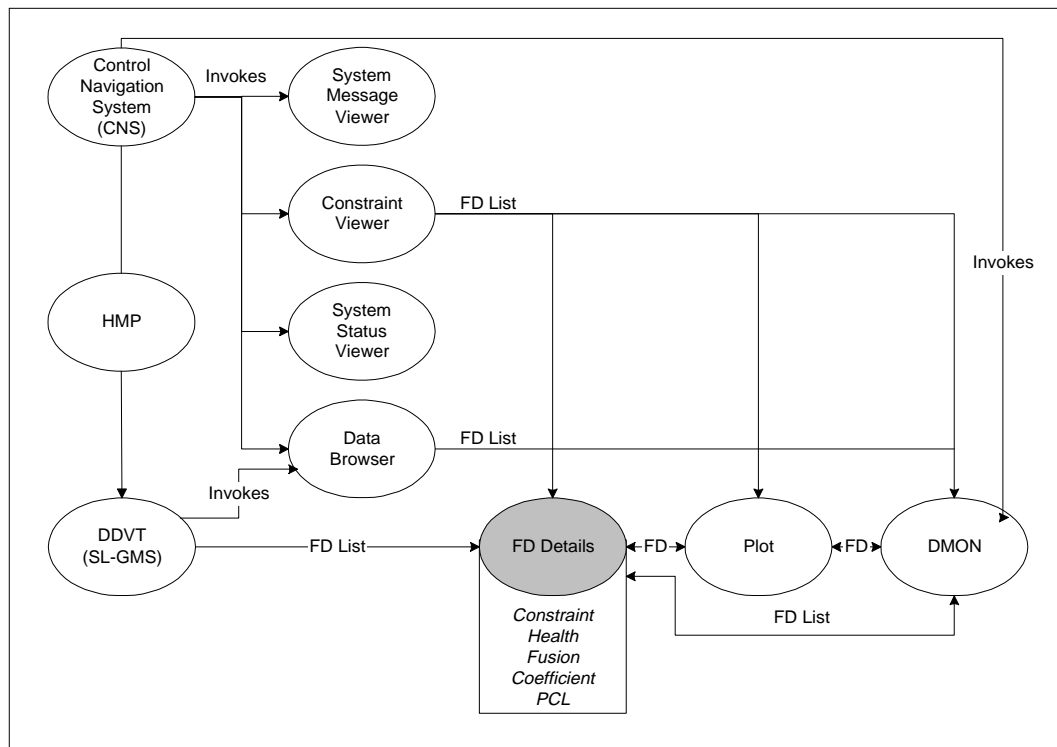
## 7. Constraint Details

### 7.1 Constraint Details Viewer

#### 7.1.1 Constraint Details Viewer

The Constraint Details Viewer is part of the FD Details Viewer. The Constraint Details Viewer is executed on the CCWS. The Constraint Details Viewer will provide the user a way to change the constraint expression.

The following diagram shows the relationship of the Constraint Details Viewer to the other components of the System Viewer:



#### 7.1.2 Constraint Details Viewer Operational Description

The Constraint Details Viewer will display detailed information about the constraint. The Constraint Details Viewer will allow the user to modify the constraints and issue a command to distribute the modification throughout the system.

### 7.2 Constraint Details Viewer Specifications

#### 7.2.1 Constraint Details Viewer Groundrules

1. Help screens will not be available to the user of Constraint Details Viewer in the Thor delivery.
2. User Display Services will provide a method to change current constraint .
3. User Display Services will provide a method to gather Time information about the constraint message.
4. The Constraint Detail Viewer will not run on the BASIS environment.

# Software Requirements and Design Specification

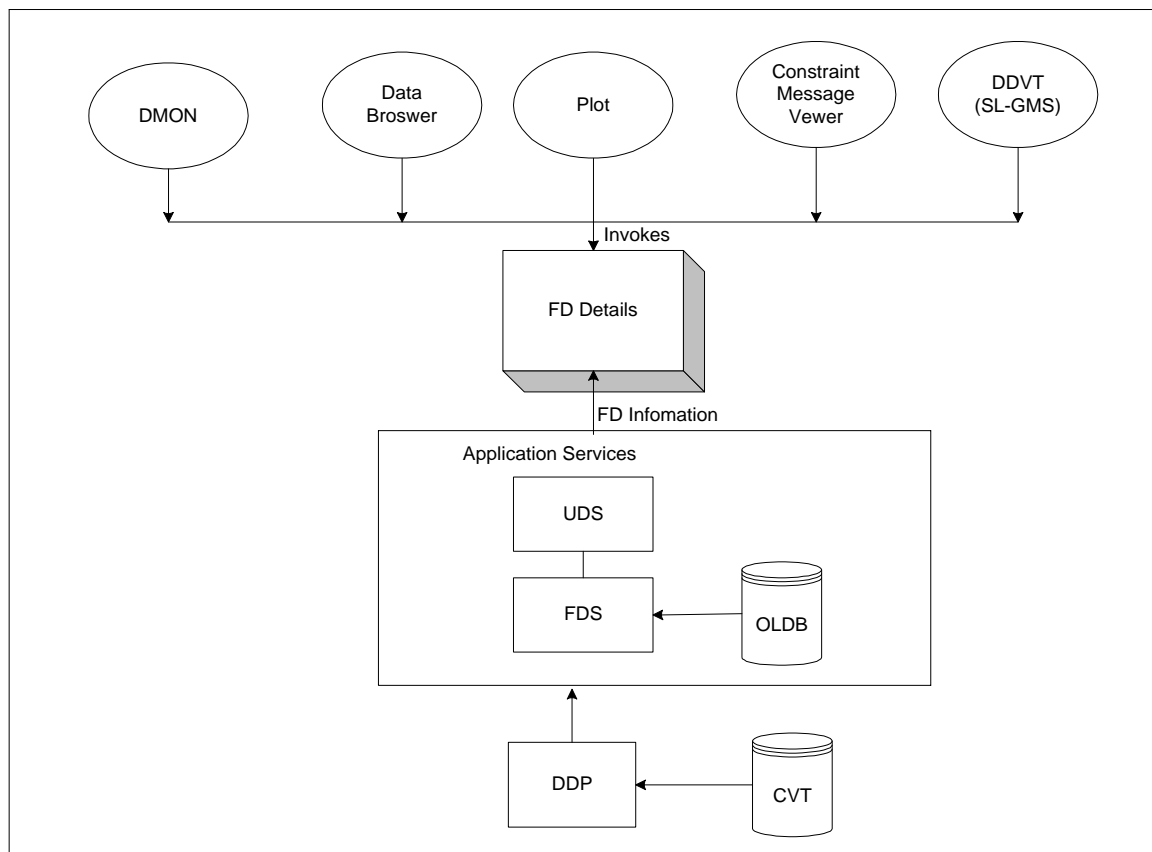
## 7.2.2 Constraint Details Viewer Functional Requirements

1. The user will be able to access the Constraint Details Viewer from the FD Details viewer. The Constraint Details Viewer will display detailed constraint information including the following:
  - Constraint ID
  - Constraint expression
  - values/health for input FDs
  - output constraint state including:
    - time of last out transition
    - time of last in transition
    - current state time
2. The Constraint Details Viewer will provide the ability to change the current constraint.
3. The Constraint Details Viewer will provide the ability to issue the change current constraint command.
4. The Constraint Details Viewer will conform to the “CLCS HCI Style Guide and Standards” documentation number 84K00230.

## 7.2.3 Constraint Details Viewer Performance Requirements

The constraint Details viewer will display within two seconds.

## 7.2.4 Constraint Details Name Interfaces Data Flow Diagrams



# Software Requirements and Design Specification

## 7.3 Constraint Details Viewer Design Specification

The Constraint Details Viewer is part of the FD Details viewer. The Constraint Details viewer allows the user to see all the constraints that are associated with one FD. The Constraint Details viewer also allows the following functionality: Assert, Alter, View and Release. The user can only modify constraint that the user owns.

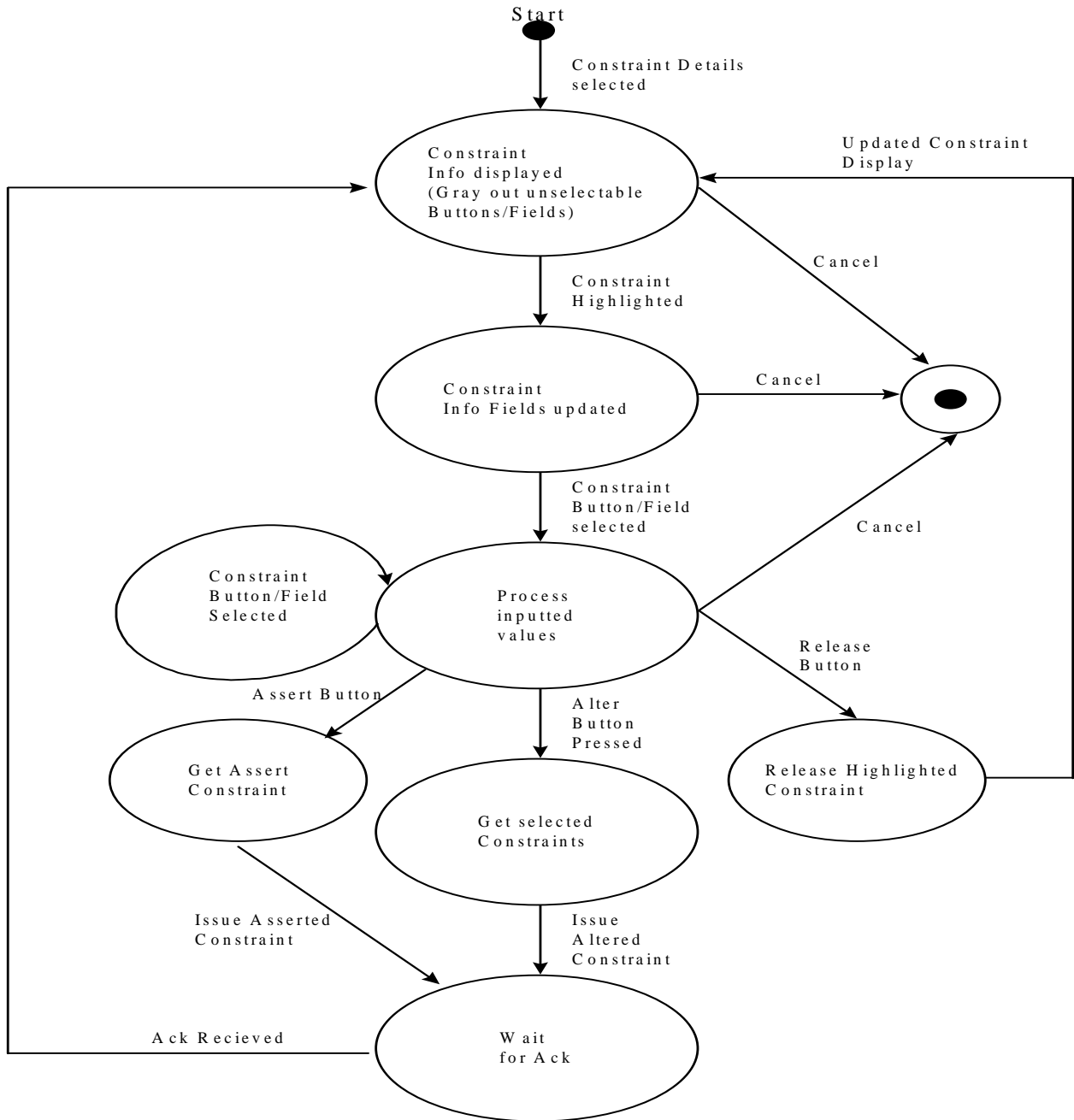
The Constraint Details class diagram is contained in the FD Viewer class diagram in appendix A.

### 7.3.1 Constraint Details Viewer State Diagram

This state diagram provides a pictorial representation of the relationship between external sources and destinations and the major and minor functions of the Constraint Details Viewer. This is an example detailed state diagram.

#### Constraint Details State Diagram

## Software Requirements and Design Specification



### 7.3.2 Constraint Details External Interfaces

#### 7.3.2.1 Constraint Details Message Formats

This data is the System Messages output by the Constraint Details Viewer

\*\*\*

Message Number = Specified system message number constant

Message Group = SSV

Severity = Error



# Software Requirements and Design Specification

Enter a Low and High value for the Analog constraint.

## **Help Information Content:**

The user must enter a Low and High constraint in the fields in order to add or update a analog constraint.

## **Details Information:**

If no value is entered, the constraint display will reject the discrete constraint and display a message.

\*\*\*

Message Number = Specified system message number constant

Message Group = SSV

Severity = Error

A constraint from the Constraint List must be highlighted in order to display the constraint information.

## **Help Information Content:**

To highlight the constraint place the arrow on top of the desired constraint a press mouse button number 1.

## **Details Information:**

No constraint information will be displayed unless the desired constraint is highlighted.

\*\*\*

Message Number = Specified system message number constant

Message Group = SSV

Severity = Error

Choose the notify type that will be displayed by the constraint manager.

## **Help Information Content:**

The following items are the Notify types available:

Transition In - The constraint viewer will only display messages that transition into range.

Transition Out - The constraint viewer will only display messages that transition out of range.

Both - The constraint viewer will display messages for all transitions.

## **Details Information:**

One of the Notify type must be checked in order to Add a constraint. The notify radio default is Both.

# Software Requirements and Design Specification

## 7.3.2.2 Constraint Details Display Formats

This is the design of Displays produced by the Constraint Details Viewer.

The screenshot shows a window titled "FD Details" with a table of constraint data and several control buttons.

| FD Name   | FD Nomenclature        | Value | Units | JTOY         |
|-----------|------------------------|-------|-------|--------------|
| H42X1120X | OX HE ISO VLV A OP IND | OPEN  |       | 307:14:30/20 |

| FDID | RSYS | Health  | Type | Subtype |
|------|------|---------|------|---------|
| 6879 | HMF  | INVALID | NONE | NONE    |

Buttons: Refresh, Constraint Viewer, Add Another FD: [text box], DMON, Plot, Print, Cancel, Help.

Constraint List

| Constraint ID | Constraint Owner | Application Att. | Transition | Digital Pattern | Discrete | Low | High |
|---------------|------------------|------------------|------------|-----------------|----------|-----|------|
|---------------|------------------|------------------|------------|-----------------|----------|-----|------|

Constraint Configuration

| Transition  | Discrete   | Digital Pattern | Low        | Analog     | High       |
|---|------------|-----------------|------------|------------|------------|
| <input checked="" type="checkbox"/> In <input type="checkbox"/> Out <input type="checkbox"/> Both | [text box] | [text box]      | [text box] | [text box] | [text box] |

Buttons: Assert, Release, Alter, Cancel.

Constraint Details Functions:

1. Constraint List will display all the constraints associated with the current FD. The constraint list displays the following information: Constraint ID, Constraint Owner, Application Attribute, Notify Type, Digital Pattern, Discrete, Low, High.
2. Notify is the type constraint transition that will be displayed by Constraint Viewer. The notify types that are available are:
  - Transition In - Displayed by constraint viewer when the constraint viewer transitions in.
  - Transition Out - Displayed by constraint viewer when the constraint viewer transitions out.
  - Both - All constraints transitions will be displayed.
3. Discrete Field - Allows the user to enter the discrete value.
4. Digital Pattern - Allows the user to create a constraint to monitor digital pattern.
5. Analog Low and High - Allows the user to enter the analog data in the fields.
6. Assert - Issues a command to add a new constraint to the FD currently displayed.
7. Alter - Issues a command to delete the highlighted constraint.
8. Release - Issues a command to update the constraint currently highlighted.

## 7.3.2.3 Constraint Details Input Formats

Not Applicable.

## 7.3.2.4 Constraint Details Recorded Data

Not Applicable.

## 7.3.2.5 Constraint Details Printer Formats

Not Applicable.

## 7.3.2.6 Interprocess Communications (C-to-C Communications?)

Not Applicable.

## 7.3.2.7 Constraint Details External Interface Calls (e.g., API Calling Formats)

This is the data that is sent between User Display Services (Application Services) via a calling mechanism (e.g., API call)

The Constraint Details Viewer requires the following Java/C++ APIs:

- An API to send commands for the following functions: Assert, Alter and Release a Discrete or Analog constraint.
- An API to send a erase command.
- An API to read all constraints associated with a FD.

## 7.3.2.8 Constraint Details Viewer Table Formats

Not Applicable.

## 7.3.3 Constraint Details Viewer Test Plan

The Constraint Details system level test environment may run in either IDE or SDE, or both. These tests are run on the CCWS platform.

1. Verify that the Constraint Details Viewer is displayed on the screen.
2. Verify that the Constraint Details Viewer list displays all the constraints associate with that FD.
3. Verify that the desired constraint information can be displayed by double clicking on a desired constraint. The following information will be displayed.
  - Constraint ID
  - Current State Time
  - Current Value
  - Health
  - Last In Transition
  - Last Out Transition
  - Notify Type
  - Discrete or Analog Low High
4. Verify that the user can add a new constraint.
5. Verify that the user can Update a current constraint that he created.
6. Verify that the user can delete a current constraint that he created.
7. Verify that the correct type of data has been enter in the Discrete or the Analog Low/High fields.
8. Verify that the Constraint Details Viewer notify type has been picked.
9. Verify that the slide bars on the Constraint Details Viewer list can be moved in order to scroll through all the data that is contained in the list.
10. Verify that a printout of the Constraint Details Viewer is printed when the print button is pressed.

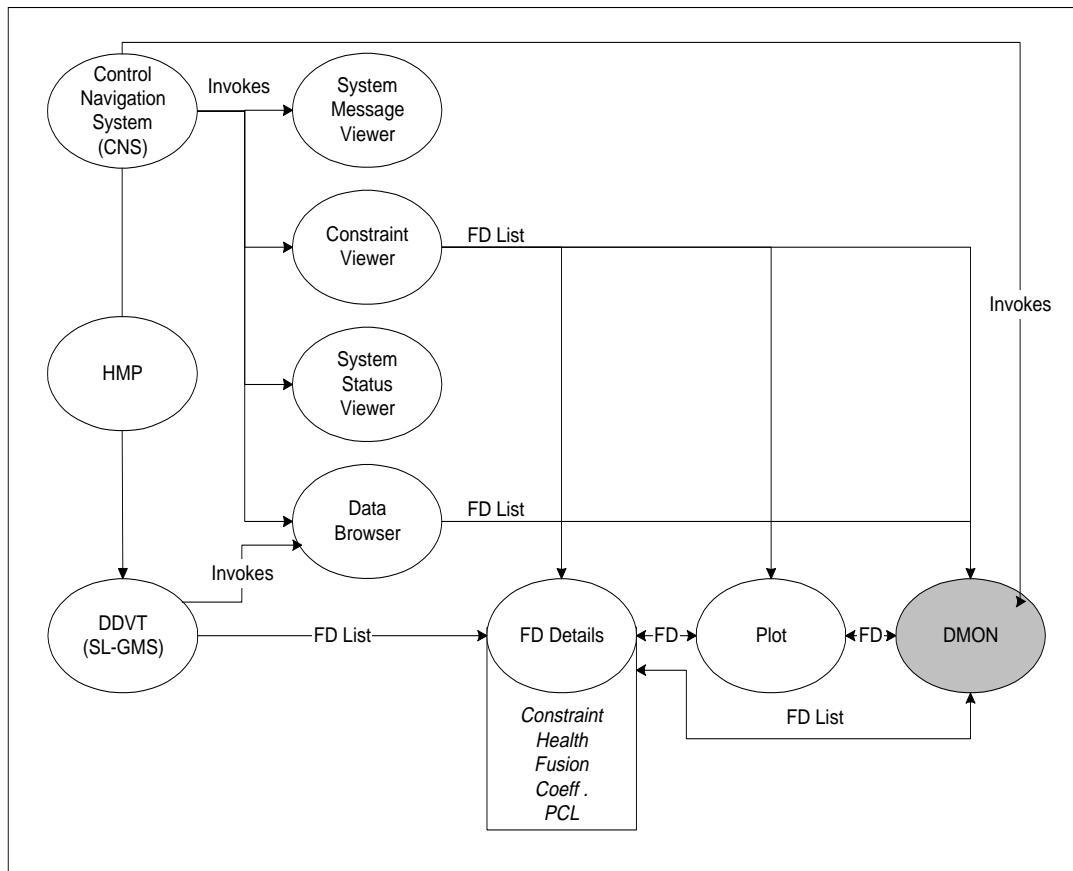
## 8. DMON Viewer

### 8.1 DMON Viewer Introduction

#### 8.1.1 DMON Viewer Overview

The DMON Viewer provides the Checkout and Launch Control System (CLCS) with a capability of providing facility to examine value, units, health, constraint, and time of update for one or more FDs.

The diagram below shows the relationship of the DMON Viewer to other components within the System Viewer CSCI.



#### 8.1.2 DMON Viewer Operational Description

The DMON Viewer is started from the DMON Selection display. The DMON Selection display is a GUI that allows the user to select FDs to display in the DMON. The DMON Selection display can also be invoked from PLOT Viewer, FD Details Viewer, and Constraint Viewer.

Once the DMON button is pressed on the above mentioned viewers, a DMON Viewer Selection Window appears. The DMON Viewer will display all the information for each FD selected in a tabular format.

## 8.2 DMON Viewer Specifications

### 8.2.1 DMON Viewer Ground Rules

1. Application Services will provide an API(s) to return value, units, health, constraint, FD name and nomenclature.
2. DMON Viewer is invoked from either Constraint Viewer, Data Browser, Plot, or FD Details.
3. For the Thor delivery the list of FDs that are displayed for selection will be a subset of all possible FDs. (e.g. User Class)
4. In order to run on the BASIS environment the On-Line Data Bank and System Configuration Table must be available. The BASIS environment will provide monitoring only capability.

### 8.2.2 DMON Viewer Functional Requirements

1. The DMON Viewer will provide a selection list of FDs.
2. The DMON Viewer will provide the ability to add and delete FDs from the selection list.
3. The DMON Viewer will provide the ability to type in an FD using the keyboard.
4. The DMON Viewer will provide the window displaying all the information for each FD selected in a tabular format.
5. The DMON Viewer will provide the following data for each FDs:
  6. FD Name
  7. Nomenclature
  8. Value
  9. Units
  10. Health
  11. Constraint
  12. Time of update
13. The DMON Viewer will cyclically update the FD Information every second.
14. The DMON Viewer will provide the ability to print the information from the current window.
15. ~~The~~ DMON Viewer will provide the ability to invoke the FD Details Viewer.
16. ~~The~~ DMON Viewer will provide the ability to invoke the Plot Viewer.
17. ~~The~~ DMON Viewer will provide the ability to also save printed text data into a file.
18. ~~The~~ DMON Viewer will provide the ability to use the Help Information.
19. ~~The~~ DMON Viewer will also execute in the ~~BASIS Basis~~ environment.
20. ~~The DMON Viewer will conform to the "CLCS HCI Style Guide & Standards" document number 84K00230.~~ The DMON Viewer will conform to the "CLCS HCI Style Guide and Standards" document number 84K00230.

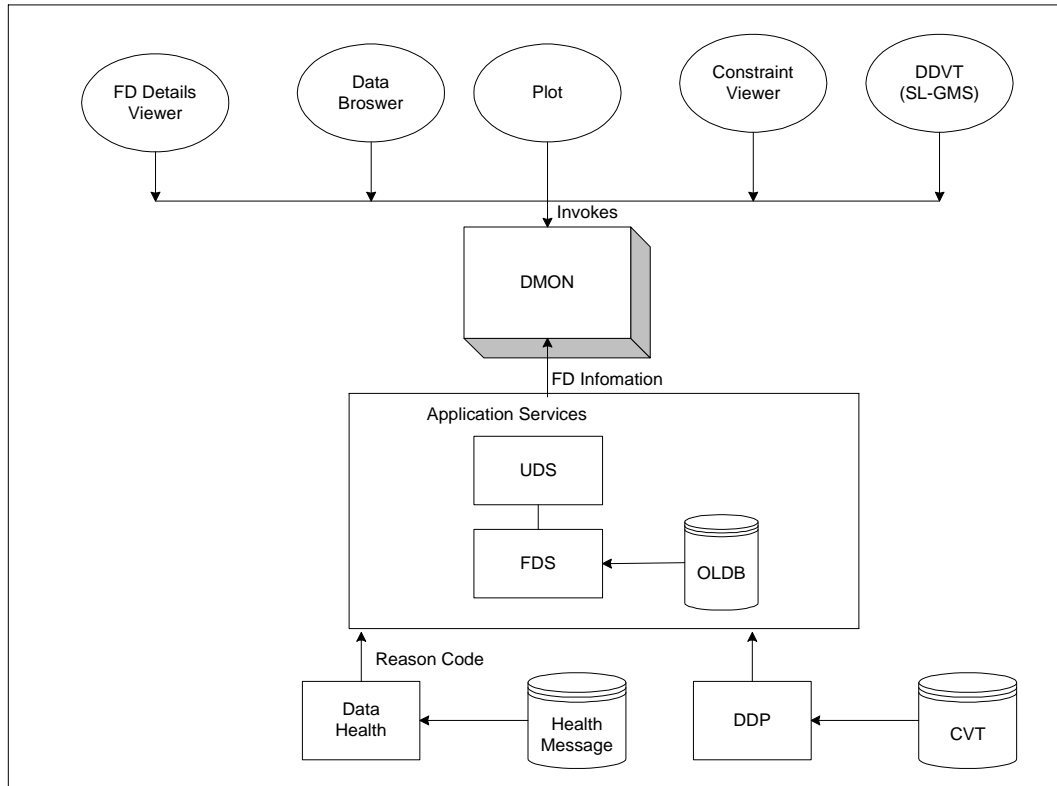
### 8.2.3 DMON Viewer Performance Requirements

- The DMON Viewer will appear on the screen within two seconds.

### 8.2.4 DMON Viewer Interfaces

The DMON Viewer interfaces with User Application Services to receive DMON information from the CVT and the OLDB. DMON is requested from the Data Browser, Plot Viewer, FD Details, or Constraint Viewers.

## 8.2.5 DMON Viewer Data Flow Diagram



The DMON Viewer requests and receives information from User Display Services.

## 8.3 DMON Viewer Design Specification

The DMON Viewer runs in the CCWS, and is invoked from the DDVT (SL-GMS Application) or from one of the following: the CNS, the FD Details Viewer, the Data Browser, the Constraint Viewer or the Plot Viewer. It is implemented in Java. Each element of variable information on the DMON Viewer screen is represented as a text field. The DMON Viewer can only be invoked once.

Appendix A contains a Class Diagram for the DMON Viewer.

### 8.3.1 DMON Viewer State Diagram

The DMON Viewer State Diagram shown in Figure 4 relates the states and the events of the DMON Viewer.

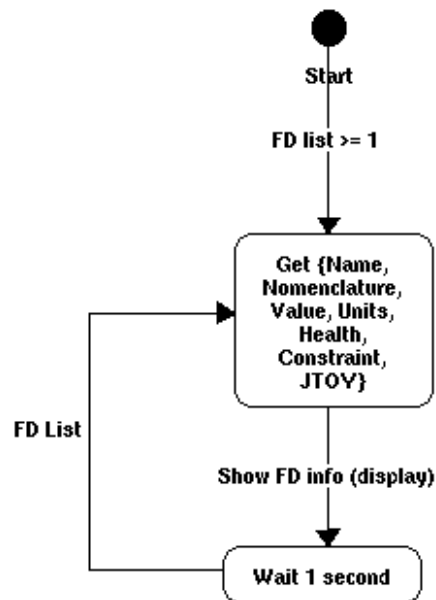


Figure 4. DMON Viewer State Diagram.

### 8.3.2 DMON Viewer External Interface

The DMON Viewer will not provide any APIs. It will call an API from the Data Handler within User Display Services. The Data Handler in UDS will send the updates to the DMON Viewer including FD Name, FD Nomenclature, Value, Units, Health, Constraint and Julian Time of Day (JTOY) . These interfaces are described in the User Display Services Interface Description Document (IDD) 84K00361.

#### 8.3.2.1 DMON Viewer Message Formats

Not applicable.

## 8.3.2.2 DMON Viewer Display Formats

The DMON Viewer is invoked via the DDVT (SL-GMS Application) or CNS. The DMON Viewer can also be invoked from the FD Details Viewers, the Data Browser, the Constraint Viewer or the Plot Viewer.



**Figure 6. Main Window of DMON Viewer.**

### Main Window Functions

1. Display the FD Name, FD Nomenclature, Value, Units, Health, Constraint and JTOY at the appropriate location listed in a window.
2. After twelve or more FDs are displayed in the main window a scroll bar will appear and the list will scroll.
3. Allow users to invoke other Viewers from the window including FD Details and Plot.
4. Allow the user to print the entire window with the print button.
5. Allow the user to close the DMON Window with the cancel button.
6. Allow the user to invoke the help window with the help button.

## 8.3.2.3 DMON Viewer Input Formats

Not applicable.

## 8.3.2.4 DMON Viewer Recorded Data

Not applicable.



# Software Requirements and Design Specification

## 8.3.2.5 DMON Viewer Printer Formats

Not applicable. Only print screens will be accomplished with the DMON Viewer.

## 8.3.2.6 DMON Viewer Inter-process Communications

The DMON Viewer receives data only from Application Services. The DMON Viewer will display the FD Name, FD Nomenclature, Value, Units, Constraint and JTOY.

## 8.3.2.7 DMON Viewer External Interface Calls

The DMON Viewer will not provide any APIs. It will call the following API from the Data Handler within User Display Services. The Data Handler in UDS will send the updates to the DMON Viewer. These interfaces are described in the User Display Services Interface Description Document (IDD) 84K00361.

## 8.3.3 DMON Viewer Test Plan

The DMON Viewer system level tests may be run in either the IDE or SDE, or both. These tests are run on the CCWS platform. Application services will be required to send data to the DMON Viewer for FD Name, FD Nomenclature, Value, Units, Health Constraint and JTOY.

The specific test cases that will be run include:

1. Verify a FD Name, FD Nomenclature, Value, Units, Health Constraint and JTOY are received in the DMON Window.
2. Verify print screen capability.
3. Verify the FD Details Button invokes the FD Details Viewer.
4. Verify the Plot Button invokes the Plot View.
5. Verify the Cancel Button closes the Data Browser.
6. *Verify the Help Button invokes the Help Window.*

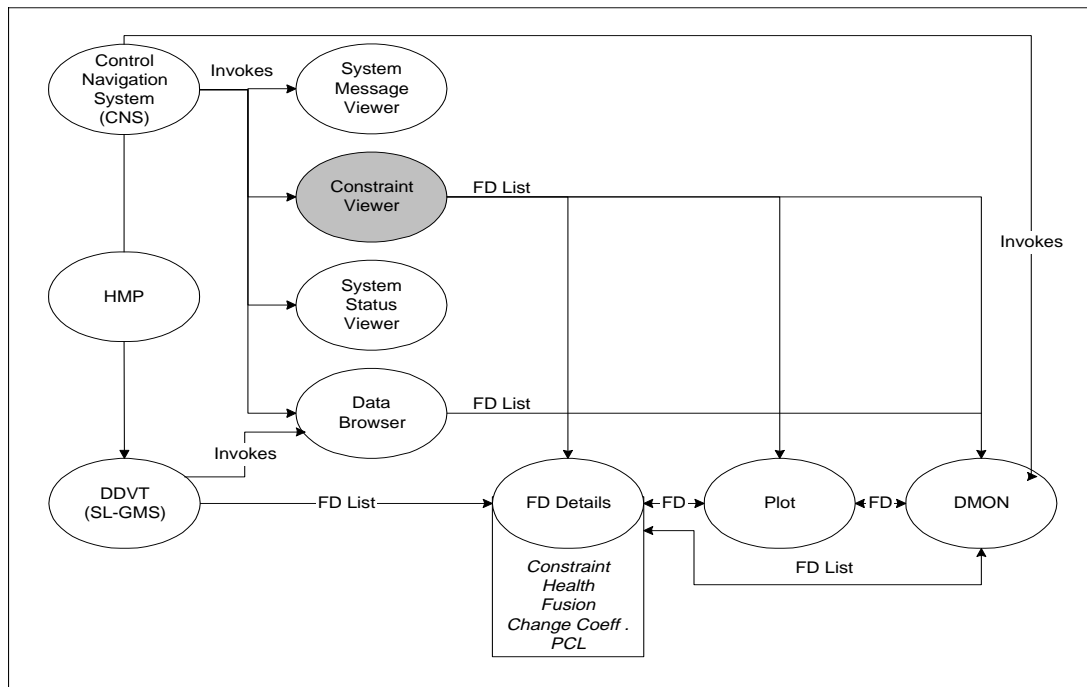
## 9. Constraint Viewer

### 9.1 Constraint Viewer

#### 9.1.1 Constraint Viewer

The Constraint Viewer is executed on the CCWS. The Constraint Viewer is invoked from the Control Navigation System (CNS) task bar. When a transition is encountered, a transition message will be displayed by the Constraint Viewer.

The following diagram shows the relationship of the Constraint Viewer to the other components of the System Viewer:



#### 9.1.2 Constraint Viewer Operational Description

The Constraint Viewer will use the first user class entry that is displayed on the CNS user class list as the default user class. The Constraint Viewer will invoke the Subscribe screen to subscribe to additional user classes. The constraints will be displayed if a transition takes place. The user can access the following viewers from the Constraint Viewer: FD Details, DMON and Plot. The user can issue the following actions on the Constraint Viewer: Print, Acknowledge, Clear, Subscribe, Cancel, History and Help. The following constraint status information will be displayed on the viewer: Number of Total Constraint and the Number of Constraint that were Acknowledged.

## 9.2 Constraint Viewer Specifications

### 9.2.1 Constraint Viewer Groundrules

1. The user will not be able to Save the user class configuration to a file.
2. The user will not be able to Load the user class configuration from a file.
3. Help screens will not be available to the user of Constraint Viewer in Thor.
4. The user will not be able to create complex constraints.
5. User Display Services will provide a method to notify the constraint viewer of transitions.
6. The Constraint Viewer will not run on the BASIS environment

### 9.2.2 Constraint Viewer Functional Requirements

#### *Constraint Viewer Definitions*

- Subscribe Sign up to monitor additional User Classes.
- Clear Acknowledge and remove the constraint message.
- Acknowledge Changes status of the constraint.
- Deselect Acknowledge, Clear and Inhibit the constraint message.
- Cancel Close the Constraint Viewer display without applying the changes.
- Time Transition time of the constraint message.

1. The Constraint Viewer will provide the user the ability to Acknowledge the constraint message.
2. The Constraint Viewer will provide the user the ability to Clear the constraint message that will acknowledge and remove the constraint message.
3. The Constraint Viewer will provide the user the ability to Deactivate the constraint message that will acknowledge, remove and Inhibit the constraint message.
4. The Constraint Viewer will provide a popup with print options to print the display or list.
5. The Constraint Viewer will provide the ability to clear multiple constraint at the same time.
6. The Constraint Viewer will highlight the constraint until it is acknowledged.
7. The Constraint Viewer will display multiple user constraints that are associated with one FD.
8. The Constraint Viewer will provide access to the following viewers:
  - FD Details
  - DMON
  - Plot viewer
9. The Constraint Viewer will provide the following commands:
  - Print
  - Acknowledge
  - Clear
  - Subscribe
  - History
  - Cancel
  - Help
10. The Constraint Viewer will filter constraints by User Class.
11. The Constraint Viewer will conform to the "CLCS HCI Style Guide and Standards" documentation number 84K00230.
12. The Constraint Viewer will provide the ability to retrieve history data by User Class.
13. The Constraint Viewer will display the constraint transition in time order by default.

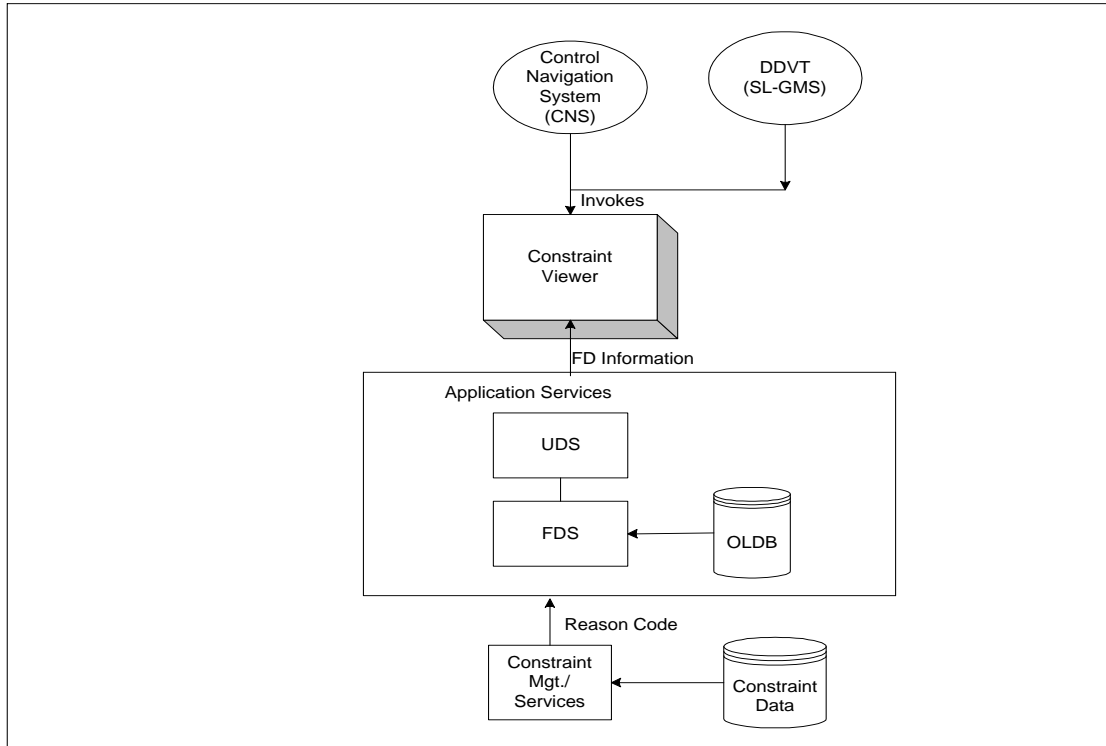
# Software Requirements and Design Specification

14. Displayed constraints will occupy a maximum of two lines and will contain the following:
  - Time of Occurrence
  - FD Name
  - Constraint ID
  - FD Nomenclature
  - Detected Value
  - Current Value
  - Health
  - Expression
  - Number of messages read
  - Total number of messages on the list
15. The Constraint Viewer will display a constraint when a transition takes place.
16. The Constraint Viewer will allow the operator to assert and release constraints for viewer notification.
17. The Constraint Viewer will update the Current Value field.
18. *The Constraint Viewer will summarize (Complex Constraint) constraint categories, constraint descriptor, RCL indicator, constraint transition state and time of occurrence.*
19. *User defined constraints can be loaded from a pre-defined Constraint File to be loaded.*
20. *User defined constraints can be saved to Constraint File.*
21. *The Constraint Viewer will provide the ability to filter constraints by FD.*
22. *The Constraint Viewer will provide the ability to retrieve history data by FD.*
23. *An alarm can be optionally requested.*
24. *The Constraint Viewer list will provide the following functionality:*
  - *Sort by Time*
  - *Sort by FD Name*
25. *The Constraint Viewer will organize the constraint by summary of constraint category to a maximum of three levels upon selection by the user.*
26. *The Constraint Viewer will provide one or all Summary Constraint Categories to be collapsed or expanded. Provide an indication of changes to constraints in collapsed categories.*
27. *The Constraint Viewer will display only those constraints corresponding to locally selected Summary Constraint Categories.*
28. *The Constraint Viewer will provide a Summary of Constraint Categories that can be added or removed locally for the duration of CCWS RSYS role assignment.*
29. *When a user subscribes to a user class the most recent messages in the history cache shall be displayed.*

## 9.2.3 Constraint Viewer Performance Requirements

The Constraint Viewer will display within two seconds.

## 9.2.4 Constraint Viewer Interfaces Data Flow Diagrams



## 9.3 Constraint Viewer Design Specification

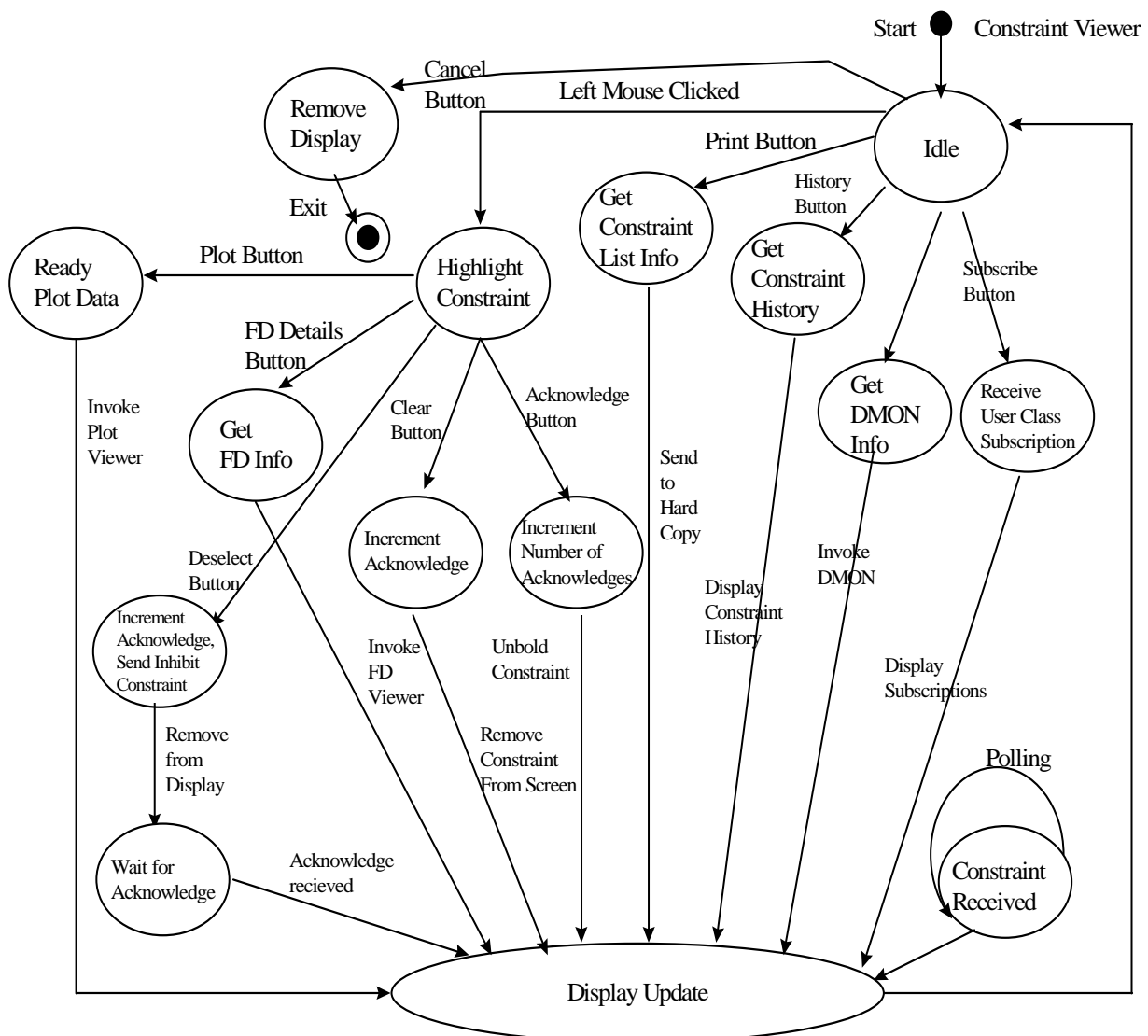
The Constraint Viewer will notify the user when a constraint has transitioned to a new state. The Constraint Viewer will allow the user to display all the constraint messages from the subscribed user classes. The user will be allowed to manipulate the constraints using the following commands: Acknowledge, Clear and Deselect. The Constraint Viewer will also retrieve history data. The user will not be allowed to manipulate constraints from the history display. The Constraint Viewer will also invoke the following viewers: FD Details, DMON, Plot and Print.

The Constraint Details class diagram is contained in the FD Viewer class diagram in appendix A.

### 9.3.1 Constraint Viewer State Diagram

This state diagram provides a pictorial representation of the relationship between external sources and destinations and the major and minor functions of the Constraint Viewer.

**Figure 3. Constraint Viewer State Diagram**



## 9.3.2 Constraint Viewer External Interfaces

### 9.3.2.1 Constraint Viewer Message Formats

This data is the System Messages output by the Constraint Viewer

\*\*\*

Message Number = Specified system message number constant

Message Group = SSV

Severity = Error

The constraint must be highlighted to perform the following actions: Acknowledge, Clear or Deselect.

#### **Help Information Content:**

The user must highlight the constraint to Acknowledge, Clear or Deselect.

#### **Details Information:**

If no constraint is highlighted, the user will not be able to perform the following functions: Acknowledge, Clear or Deselect.

\*\*\*

Message Number = Specified system message number constant

Message Group = SSV

Severity = Error

Unable to print request to the device.

#### **Help Information Content:**

The printer is not connected to the network.

#### **Details Information:**

If the application can not find the device, the submitted print job will be unable to complete.

\*\*\*

Message Number = Specified system message number constant

Message Group = SSV

Severity = Error

Unable to invoke FD Details.

#### **Help Information Content:**

The FD Details can not be invoked from the Constraint Message Viewer.

#### **Details Information:**

Determine if the FD Details viewer can be invoked from another viewer.

\*\*\*

Message Number = Specified system message number constant

Message Group = SSV

Severity = Error

Unable to invoke DMON.

# Software Requirements and Design Specification

## **Help Information Content:**

The DMON can not be invoked from the Constraint Message Viewer.

## **Details Information:**

Determine if the DMON viewer can be invoked from another viewer.

\*\*\*

Message Number = Specified system message number constant

Message Group = SSV

Severity = Error

Unable to invoke Plot.

## **Help Information Content:**

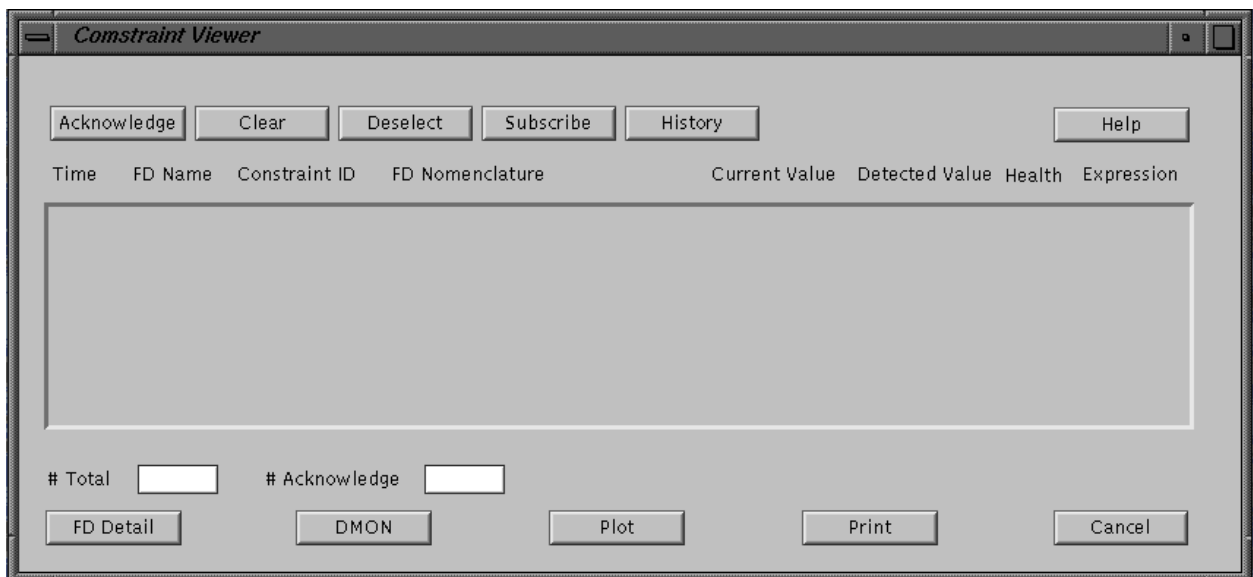
The Plot Details can not be invoked from the Constraint Message Viewer.

## **Details Information:**

Determine if the Plot viewer can be invoked from another viewer.

### **9.3.2.2 Constraint Viewer Display Formats**

This is the design of displays produced by the Constraint Viewer.



Constraint Viewer Functions:

1. Acknowledge allows the user to changes the status of the constraint.
2. Clear allows the user to remove the constraint. Clear will Acknowledge and remove the constraint message.
3. Deselect allows the user to remove constraint from being monitored. Deselect will Acknowledge, Clear and Inhibit the constraint message.
4. Subscribe allows the user to sign up to monitor additional user class.
5. History allows the user to display Historical constraint data.
6. Constraint message will contain the following information: Time, FD Name, Constraint ID, FD Nomenclature, Current Value, Detected Value, Health and Expression.
7. FD Details button will invoke the FD Details display.
8. DMON button will invoke the DMON display.
9. Plot button will invoke the plot display.



# Software Requirements and Design Specification

10. Print will print a hardcopy.
11. Cancel allows the user to close the Constraint Viewer display without applying the changes.

## 9.3.2.3 Constraint Viewer Input Formats

Not Applicable.

## 9.3.2.4 Constraint Viewer Recorded Data

Not Applicable.

## 9.3.2.5 Constraint Viewer Printer Formats

The Constraint Viewer provides print constraint list, Number of total messages, Number of Acknowledged messages and subscriptions.

## 9.3.2.6 Interprocess Communications (C-to-C Communications?)

Not Applicable.

## 9.3.2.7 Constraint Viewer External Interface Calls (e.g., API Calling Formats)

This is the data that is sent between User Display Services (Application Services) via a calling mechanism (e.g., API call)

The Constraint Viewer requires the following Java/C++ APIs:

- An API to send commands for the following functions: Assert, Update and Remove a Discrete or Analog constraint.
- An API to send a Erase command.
- An API to send a Subscribe command.
- An API to read the constraint History cache.
- An API to notify the Constraint Viewer.

## 9.3.2.8 Constraint Viewer Table Formats

Not Applicable.

## 9.3.3 Constraint Viewer Test Plan

The Constraint Viewer system level test environment may run in either IDE or SDE, or both. These tests are run on the CCWS platform.

1. Verify that the Constraint Viewer is displayed on the screen.
2. Verify that the Constraint Viewer list displays all the constraints.
3. Verify that the desired constraint information can be displayed by highlighting the desired constraint and pressing the FD Details button. The following information will be displayed.
  - Constraint ID
  - Current State Time
  - Current Value
  - Health
  - Last In Transition
  - Last Out Transition
  - Notify Type
  - Discrete or Analog Low High

## Software Requirements and Design Specification

4. Verify that the following constraint information is displayed on the Constraint Viewer:
  - Time
  - FD Name
  - Constraint ID
  - Constraint Descriptor
  - Current Value
  - Detected Value
  - Health
  - Expression
  - Total Number of constraint messages
  - Total number of Acknowledged constraint messages
5. Verify that the user can invoke the following displays:
  - FD Details
  - DMON
  - Plot
  - Print
6. Verify that the user can highlight a constraint and press the Plot button to display the FD graph.
7. Verify that the slide bars on the Constraint Viewer list can be moved in order to scroll through all the data that is contained in the list.
8. Verify that a printout of the Constraint Details Viewer is printed when the print button is pressed.
9. Verify that the user can Acknowledge a constraint message.
10. Verify that the user can Clear a constraint message.
11. Verify that the user can Deselect a constraint message.
12. Verify that the user can Subscribe to a User Class to display the constraint messages associated the user class.
13. Verify that the user can display the constraint message History.
14. Verify that the correct value is in the total number of constraint messages window.
15. Verify that the correct value is in the total number of acknowledged constraint messages window.

# Software Requirements and Design Specification

## APPENDIX A

| <b>Data Browser</b>        |
|----------------------------|
| FD Name<br>FD Nomenclature |
| get_data_from_UDS          |

Figure A-1. Class Diagram for Data Browser

# Software Requirements and Design Specification

The following is the FD Viewer Class Diagram. The FD Viewer includes the FD Details Viewer, Constraint Viewer, Health Viewer, Data Fusion Viewer, Coefficient Viewer, and PCL Viewer.

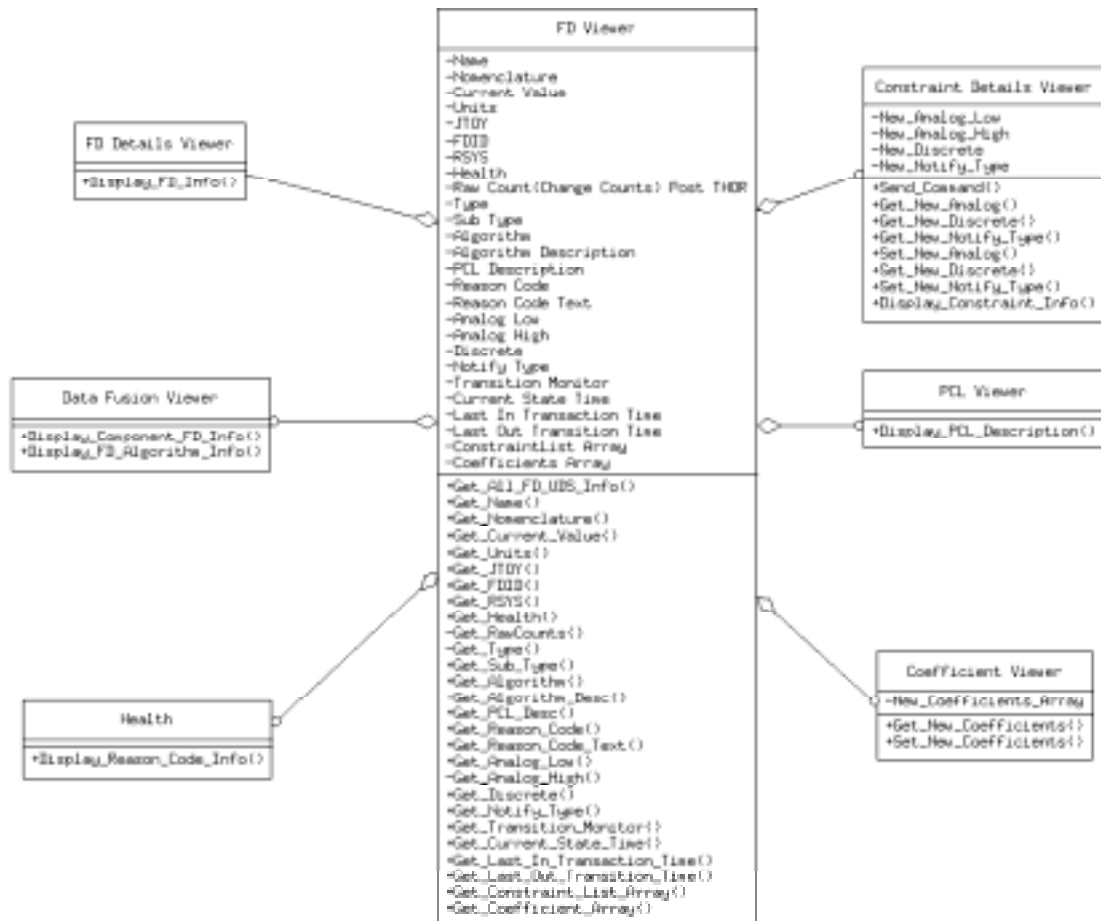


Figure A-2. FD Viewer Class Diagram

# Software Requirements and Design Specification

| DMON   |
|--|
| FD Name<br>FD Nomenclature<br>Value<br>Units<br>Health<br>Constraint<br>JTOY |
| Get_DMOM_info(FD)<br>Show_DMOM_info(FD)                                      |

Figure A-3. Class Diagram for DMON Viewer

| Constraint Viewer  |
|--|
| - Constraint List Array<br>- totalNumAcknowledged<br>- totalNumMessages<br>- index   |
| + getHighlight(index)<br>+ getTotalAcknowledged()<br>+ setTotalAcknowledge()<br>+ getTotalMessages()<br>+ setTotalMesages()<br>+ incrementNummessages()<br>+ invokeFD_Details()<br>+ invokeDMON()<br>+ invokePlot()<br>+ acknowledgeEventHandler()<br>+ clearEventHandler()<br>+ desselectEventHandler()<br>+ SubscribeEventHandler()<br>+ historyEventHandler() |

Figure A-4. Class Diagram for Constraint Viewer